FORMALIZED MATHEMATICS
Vol. 28, No. 2, Pages 187–196, 2020
DOI: 10.2478/forma-2020-0016

sciendo
https://www.sciendo.com/

Partial Correctness of a Fibonacci Algorithm

Artur Korniłowicz
Institute of Informatics
University of Białystok
Poland

Summary. In this paper we introduce some notions to facilitate formulating and proving properties of iterative algorithms encoded in nominative data language [19] in the Mizar system [3], [1]. It is tested on verification of the partial correctness of an algorithm computing n-th Fibonacci number:

```
i := 0
s := 0
b := 1
c := 0
while (i <> n)
   c := s
   s := b
   b := c + s
   i := i + 1
return s
```

This paper continues verification of algorithms [10], [13], [12] written in terms of simple-named complex-valued nominative data [6], [8], [17], [11], [14], [15]. 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 [16], [18], [7], [5].

MSC: 68Q60 03B70 03B35

Keywords: nominative data; program verification; Fibonacci sequence

MML identifier: NOMIN_7, version: 8.1.10 5.63.1382

1. Introduction

From now on D denotes a non empty set, m, n, N denote natural numbers, z_2 denotes a non zero natural number, f_1, f_2, f_3, f_4, f_5, f_6 denote binominative functions of D, p_1, p_2, p_3, p_4, p_5, p_6, p_7 denote partial predicates of D, d, v denote objects.

Observe that V, A denote sets, z denotes an element of V, val denotes a function, loc denotes a V-valued function, d_1 denotes a non-atomic nominative data of V and A, and T denotes a nominative data with simple names from V and complex values from A.

Let R_1, R_2 be binary relations. We say that R_1 is valid w.r.t. R_2 if and only if

(Def. 1) $\operatorname{rng} R_1 \subseteq \operatorname{dom} R_2$.

Let us consider V, loc, val, and N. We say that loc and val are different w.r.t. N if and only if

(Def. 2) for every natural numbers m, n such that $1 \leqslant m \leqslant N$ and $1 \leqslant n \leqslant N$ holds $val(m) \neq loc_{/n}$.

Now we state the propositions:

(1) Suppose $loc \restriction \operatorname{Seg} N$ is one-to-one and $\operatorname{Seg} N \subseteq \operatorname{dom} loc$. Let us consider natural numbers i, j. Suppose $1 \leqslant i \leqslant N$ and $1 \leqslant j \leqslant N$ and $i \neq j$. Then $loc_{/i} \neq loc_{/j}$.

(2) If V is not empty and $v \in \operatorname{dom} d_1$, then $(d_1 \nabla_a^z (v \Rightarrow_a)(d_1))(z) = d_1(v)$.

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

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

Now we state the proposition:

(3) Unconditional composition rule for 6 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 $\langle \sim p_2, f_2, p_3\rangle$ 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. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6), p_7\rangle$ is an SFHT of D.

Let us consider V, A, loc, val, and d_1. Let z_2 be a natural number. Assume $z_2 > 0$. The functor LocalOverlapSeq(A, loc, val, d_1, z_2) yielding a finite sequence of elements of $\mathrm{ND}_{\mathrm{SC}}(V, A)$ is defined by

(Def. 4) $\operatorname{len} it = z_2$ and $it(1) = d_1 \nabla_a^{(loc_{/1})}(val(1) \Rightarrow_a)(d_1)$ and for every natural number n such that $1 \leqslant n < \operatorname{len} it$ holds $it(n+1) = it(n)\nabla_a^{(loc_{/n+1})}(val(n+1) \Rightarrow_a)(it(n))$.

Let f be a function. We say that f is (V,A)-nonatomicND yielding if and only if

(Def. 5) for every object n such that $n \in \operatorname{dom} f$ holds $f(n)$ is a non-atomic nominative data of V and A.

Let f be a finite sequence. Let us observe that f is (V,A)-nonatomicND yielding if and only if the condition (Def. 6) is satisfied.

(Def. 6) for every natural number n such that $1 \leqslant n \leqslant \operatorname{len} f$ holds $f(n)$ is a non-atomic nominative data of V and A.

Let us consider d_1. Observe that $\langle d_1 \rangle$ is (V,A)-nonatomicND yielding and there exists a finite sequence which is (V,A)-nonatomicND yielding.

Now we state the proposition:

(4) Let us consider a (V,A)-nonatomicND yielding finite sequence f. If $n \in \operatorname{dom} f$, then $f(n)$ is a non-atomic nominative data of V and A.

Let us consider V, A, loc, val, d_1, and z_2. One can check that LocalOverlapSeq (A, loc, val, d_1, z_2) is (V,A)-nonatomicND yielding.

Let us consider n. Let us observe that $(\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n)$ is function-like and relation-like.

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

(5) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A. Then suppose $1 \leqslant n < z_2$ and $val(n+1) \in \operatorname{dom}((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n))$. Then $\operatorname{dom}((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n+1)) = \{loc_{/n+1}\} \cup \operatorname{dom}((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n))$.

(6) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A. Then suppose $1 \leqslant n < z_2$ and $val(n+1) \in \operatorname{dom}((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n))$. Then $\operatorname{dom}((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n)) \subseteq \operatorname{dom}((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n+1))$. The theorem is a consequence of (5).

Let us consider V, A, loc, val, d_1, and z_2. We say that loc, val and z_2 are correct w.r.t. d_1 if and only if

(Def. 7) V is not empty and V is without nonatomic nominative data w.r.t. A and val is valid w.r.t. d_1 and $\operatorname{dom}(\text{LocalOverlapSeq}(A, loc, val, d_1, z_2)) \subseteq \operatorname{dom} val$.

Now we state the proposition:

(7) Suppose loc, val and z_2 are correct w.r.t. d_1. Let us consider a natural number n. Suppose $1 \leqslant n \leqslant z_2$. Then $\operatorname{dom} d_1 \subseteq \operatorname{dom}((\text{LocalOverlapSeq}(A,$

$loc, val, d_1, z_2))(n))$.

PROOF: Set F = LocalOverlapSeq(A, loc, val, d_1, z_2). Define $\mathcal{P}$[natural number] $\equiv$ if $1 \leqslant \$_1 \leqslant z_2$, 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, $\mathcal{P}[k]$. □

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

(8) Suppose loc, val and z_2 are correct w.r.t. d_1. Then suppose $1 \leqslant n \leqslant m \leqslant z_2$. Then dom((LocalOverlapSeq$(A, loc, val, d_1, z_2))(n)) \subseteq$ dom ((LocalOverlapSeq$(A, loc, val, d_1, z_2))(m))$. The theorem is a consequence of (7) and (6).

(9) Suppose loc, val and z_2 are correct w.r.t. d_1. Then if $1 \leqslant n \leqslant m \leqslant z_2$, then $loc_{/n} \in$ dom ((LocalOverlapSeq$(A, loc, val, d_1, z_2))(m))$. The theorem is a consequence of (8) and (7).

(10) Suppose loc, val and z_2 are correct w.r.t. d_1. Then if ($n \in \operatorname{dom} val$ or $1 \leqslant n \leqslant z_2$) and $1 \leqslant m \leqslant z_2$, then $val(n) \in$ dom((LocalOverlapSeq$(A, loc, val, d_1, z_2))(m))$. The theorem is a consequence of (7).

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

(11) Suppose loc, val and z_2 are correct w.r.t. d_1 and loc and val are different w.r.t. z_2. Then suppose $1 \leqslant n \leqslant m < j \leqslant z_2$. Then ((LocalOverlapSeq$(A, loc, val, d_1, z_2))(n))(val(j))$ = (LocalOverlapSeq$(A, loc, val, d_1, z_2))(m)(val(j))$.

PROOF: Set F = LocalOverlapSeq(A, loc, val, d_1, z_2). Set $l_1 = val(j)$. Define $\mathcal{P}$[natural number] $\equiv$ if $n \leqslant \$_1 < j \leqslant z_2$, then $F(n)(l_1) = F(\$_1)(l_1)$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. □

(12) Suppose loc, val and z_2 are correct w.r.t. d_1 and $\operatorname{Seg} z_2 \subseteq \operatorname{dom} loc$ and $loc \restriction \operatorname{Seg} z_2$ is one-to-one. Then suppose $1 \leqslant j \leqslant n \leqslant m \leqslant z_2$. Then (LocalOverlapSeq$(A, loc, val, d_1, z_2))(n)(loc_{/j})$ = (LocalOverlapSeq$(A, loc, val, d_1, z_2))(m)(loc_{/j})$.

PROOF: Set F = LocalOverlapSeq(A, loc, val, d_1, z_2). Set $l_1 = loc_{/j}$. Define $\mathcal{P}$[natural number] $\equiv$ if $n \leqslant \$_1 \leqslant z_2$, then $F(n)(l_1) = F(\$_1)(l_1)$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. □

(13) Let us consider a z_2-element finite sequence val. Suppose $\operatorname{Seg} z_2 \subseteq \operatorname{dom} loc$ and $loc \restriction \operatorname{Seg} z_2$ is one-to-one and loc and val are different w.r.t. z_2 and loc, val and z_2 are correct w.r.t. d_1. If $1 \leqslant n \leqslant m \leqslant z_2$, then ((LocalOverlapSeq $(A, loc, val, d_1, z_2))(m))(loc_{/n}) = d_1(val(n))$.

Proof: Set $F = \mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2)$. Define $\mathcal{P}$[natural number] $\equiv$ if $n \leqslant \$_1 \leqslant z_2$, then $(F(\$_1))(loc_{/n}) = d_1(val(n))$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. □

(14) Let us consider a z_2-element finite sequence val. Suppose loc and val are different w.r.t. z_2 and loc, val and z_2 are correct w.r.t. d_1. Let us consider natural numbers m, n. Suppose $1 \leqslant m \leqslant z_2$ and $1 \leqslant n \leqslant z_2$. Then $((\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(m))(val(n)) = d_1(val(n))$.

Proof: Set $F = \mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2)$. Define $\mathcal{P}$[natural number] $\equiv$ if $1 \leqslant \$_1 \leqslant z_2$, then $(F(\$_1))(val(n)) = d_1(val(n))$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. □

(15) Let us consider a z_2-element finite sequence val. Suppose loc, val and z_2 are correct w.r.t. d_1 and $\mathrm{Seg}\, z_2 \subseteq \mathrm{dom}\, loc$ and $loc \restriction \mathrm{Seg}\, z_2$ is one-to-one and loc and val are different w.r.t. z_2. Let us consider natural numbers j, m, n. Suppose $1 \leqslant j < m \leqslant n \leqslant z_2$. Then $((\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n))(loc_{/m}) = (\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(j)(val(m))$.

Proof: Set $F = \mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2)$. Define $\mathcal{P}$[natural number] $\equiv$ if $m \leqslant \$_1 \leqslant z_2$, then $(F(\$_1))(loc_{/m}) = F(j)(val(m))$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. □

Let us consider V, A, loc, and val. Let z_2 be a natural number. Assume $0 < z_2$. The functor initial-assignments-Seq(A, loc, val, z_2) yielding a finite sequence of elements of $\mathrm{ND_{SC}}(V, A) \dot{\to} \mathrm{ND_{SC}}(V, A)$ is defined by

(Def. 8) $\mathrm{len}\, it = z_2$ and $it(1) = \mathrm{Asg}^{(loc_{/1})}(val(1) \Rightarrow_a)$ and for every natural number n such that $1 \leqslant n < z_2$ holds $it(n+1) = it(n) \bullet (\mathrm{Asg}^{(loc_{/n+1})}(val(n+1) \Rightarrow_a))$.

The functor initial-assignments(A, loc, val, z_2) yielding a binominative function over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 9) $(\text{initial-assignments-Seq}(A, loc, val, z_2))(z_2)$.

2. Main Algorithm

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

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

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

(Def. 11) $\mathrm{WH}(\neg\,\mathrm{Equality}(A, loc_{/1}, loc_{/3}), \mathrm{Fibonacci\text{-}loop\text{-}body}(A, loc))$.

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

(Def. 12) $\mathrm{initial\text{-}assignments}(A, loc, val, 6) \bullet (\mathrm{Fibonacci\text{-}main\text{-}loop}(A, loc))$.

Let us consider z. The functor Fibonacci-program(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. 13) $\mathrm{Fibonacci\text{-}main\text{-}part}(A, loc, val) \bullet (\mathrm{Asg}^{z}((loc_{/4}) \Rightarrow_{a}))$.

From now on n_0 denotes a natural number.

Let us consider V, A, val, n_0, and d. We say that val, n_0, and d constitute a valid input for the Fibonacci algorithm w.r.t. V and A 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 $\{val(1), val(2), val(3), val(4), val(5), val(6)\} \subseteq \mathrm{dom}\, d_1$ and $d_1(val(1)) = 0$ and $d_1(val(2)) = 1$ and $d_1(val(3)) = n_0$ and $d_1(val(4)) = 0$ and $d_1(val(5)) = 1$ and $d_1(val(6)) = 0$.

The functor valid-Fibonacci-input(V, A, val, 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 val, n_0, and d constitute a valid input for the Fibonacci algorithm w.r.t. V and A, then $it(d) = \mathit{true}$ and if val, n_0, and d do not constitute a valid input for the Fibonacci algorithm w.r.t. V and A, then $it(d) = \mathit{false}$.

One can check that valid-Fibonacci-input(V, A, val, n_0) is total.

Let us consider z and d. We say that z, n_0, and d constitute a valid output for the Fibonacci algorithm w.r.t. A if and only if

(Def. 16) there exists a non-atomic nominative data d_1 of V and A such that $d = d_1$ and $z \in \mathrm{dom}\, d_1$ and $d_1(z) = \mathrm{Fib}(n_0)$.

The functor valid-Fibonacci-output(A, z, n_0) yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by

(Def. 17) $\mathrm{dom}\, it = \{d$, where d is a nominative data with simple names from V and complex values from $A : d \in \mathrm{dom}(z \Rightarrow_{a})\}$ and for every object d such that $d \in \mathrm{dom}\, it$ holds if z, n_0, and d constitute a valid output for the Fibonacci algorithm w.r.t. A, then $it(d) = \mathit{true}$ and if z, n_0, and d do not constitute a valid output for the Fibonacci algorithm w.r.t. A, then $it(d) = \mathit{false}$.

Let us consider *loc* and d. We say that *loc*, n_0, and d constitute an invariant for the Fibonacci algorithm w.r.t. A if and only if

(Def. 18) there exists a non-atomic nominative data d_1 of V and A such that $d = d_1$ and $\{loc_{/1}, loc_{/2}, loc_{/3}, loc_{/4}, loc_{/5}, loc_{/6}\} \subseteq \operatorname{dom} d_1$ and $d_1(loc_{/2}) = 1$ and $d_1(loc_{/3}) = n_0$ and there exists a natural number I such that $I = d_1(loc_{/1})$ and $d_1(loc_{/4}) = \mathrm{Fib}(I)$ and $d_1(loc_{/5}) = \mathrm{Fib}(I+1)$.

The functor Fibonacci-inv(A, loc, n_0) yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by

(Def. 19) $\operatorname{dom} it = \mathrm{ND_{SC}}(V, A)$ and for every object d such that $d \in \operatorname{dom} it$ holds if *loc*, n_0, and d constitute an invariant for the Fibonacci algorithm w.r.t. A, then $it(d) = true$ and if *loc*, n_0, and d do not constitute an invariant for the Fibonacci algorithm w.r.t. A, then $it(d) = false$.

Let us observe that Fibonacci-inv(A, loc, n_0) is total.

Now we state the propositions:

(16) Let us consider a 6-element finite sequence *val*. Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and $\mathrm{Seg}\,6 \subseteq \operatorname{dom} loc$ and $loc \restriction \mathrm{Seg}\,6$ is one-to-one and *loc* and *val* are different w.r.t. 6. Then $\langle$valid-Fibonacci-input(V, A, val, n_0), initial-assignments$(A, loc, val, 6)$, Fibonacci-inv$(A, loc, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$.

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 $i_1 = val(1)$. Set $j_1 = val(2)$. Set $n_1 = val(3)$. Set $s_1 = val(4)$. Set $b_1 = val(5)$. Set $c_1 = val(6)$. Set $I =$ valid-Fibonacci-input(V, A, val, n_0). Set $i_2 =$ Fibonacci-inv(A, loc, n_0). Set $D_3 = i_1 \Rightarrow_a$. Set $D_4 = j_1 \Rightarrow_a$. Set $D_5 = n_1 \Rightarrow_a$. Set $D_6 = s_1 \Rightarrow_a$. Set $D_1 = b_1 \Rightarrow_a$. Set $D_2 = c_1 \Rightarrow_a$. Set $U_1 = \mathrm{S_P}(i_2, D_2, c)$. Set $T_1 = \mathrm{S_P}(U_1, D_1, b)$. Set $S_1 = \mathrm{S_P}(T_1, D_6, s)$. Set $R_1 = \mathrm{S_P}(S_1, D_5, n)$. Set $Q_1 = \mathrm{S_P}(R_1, D_4, j)$. Set $P_1 = \mathrm{S_P}(Q_1, D_3, i)$. $I \models P_1$. $\square$

(17) Suppose V is not empty and A is complex containing 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 $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and $\mathrm{Seg}\,6 \subseteq \operatorname{dom} loc$ and $loc \restriction \mathrm{Seg}\,6$ is one-to-one. Then $\langle$Fibonacci-inv(A, loc, n_0), Fibonacci-loop-body(A, loc), Fibonacci-inv$(A, loc, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (1) and (2).

(18) Suppose V is not empty and A is complex containing 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 $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and $\mathrm{Seg}\,6 \subseteq \operatorname{dom} loc$ and $loc \restriction \mathrm{Seg}\,6$ is one-to-one. Then $\langle$Fibonacci-inv(A, loc, n_0), Fibonacci-main-loop(A, loc), Equality$(A, loc_{/1},$

$loc_{/3}) \wedge$Fibonacci-inv$(A, loc, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (17).

(19) Let us consider a 6-element finite sequence val. Suppose V is not empty and A is complex containing 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 $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and $\mathrm{Seg}\,6 \subseteq \mathrm{dom}\, loc$ and $loc\restriction \mathrm{Seg}\,6$ is one-to-one and loc and val are different w.r.t. 6. Then $\langle$valid-Fibonacci-input(V, A, val, n_0), Fibonacci-main-part(A, loc, val), Equality$(A, loc_{/1}, loc_{/3}) \wedge$Fibonacci-inv$(A, loc, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (16) and (18).

(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 Equality$(A, loc_{/1}, loc_{/3}) \wedge$ Fibonacci-inv$(A, loc, n_0) \models \mathrm{S_P}$ (valid-Fibonacci-output$(A, z, n_0), (loc_{/4}) \Rightarrow_a, z)$.

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 $D_6 = s \Rightarrow_a$. Set $E_1 = \{i, j, n, s, b, c\}$. 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 there exists a natural number I such that $I = d_1(i)$ and $d_1(s) = \mathrm{Fib}(I)$ and $d_1(b) = \mathrm{Fib}(I + 1)$. Reconsider $d_3 = d$ as a nominative data with simple names from V and complex values from A. Set $L = d_3 \nabla^z_a D_6(d_3)$. z, n_0, and L constitute a valid output for the Fibonacci algorithm w.r.t. A. $\square$

(21) 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 $\langle$Equality$(A, loc_{/1}, loc_{/3}) \wedge$Fibonacci-inv$(A, loc, n_0)$, $\mathrm{Asg}^z((loc_{/4}) \Rightarrow_a$), valid-Fibonacci-output$(A, z, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (20).

(22) Suppose for every T, T is a value on $loc_{/1}$ and T is a value on $loc_{/3}$. Then $\langle\sim$ (Equality$(A, loc_{/1}, loc_{/3}) \wedge$Fibonacci-inv$(A, loc, n_0))$, $\mathrm{Asg}^z((loc_{/4}) \Rightarrow_a)$, valid-Fibonacci-output$(A, z, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$.

(23) Partial correctness of a Fibonacci algorithm:
Let us consider a 6-element finite sequence val. Suppose V is not empty and A is complex containing 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 $loc_{/2}$ 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 $\mathrm{Seg}\,6 \subseteq \mathrm{dom}\, loc$ and $loc\restriction \mathrm{Seg}\,6$ is one-to-one and loc and val are different w.r.t. 6. Then $\langle$valid-Fibonacci-input(V, A, val, n_0), Fibonacci-program(A, loc, val, z), valid-Fibonacci-output$(A, z, n_0)\rangle$ is an S-FHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (19), (21), and (22).

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 Kornilowicz, 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.

[15] Artur Kornilowicz, 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.

[16] 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.

[17] 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.

[18] 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.

[19] 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 May 31, 2020