FORMALIZED MATHEMATICS
Vol. 19, No. 4, Pages 241–250, 2011
DOI: 10.2478/v10037-011-0033-2

Representation Theorem for Stacks

Grzegorz Bancerek
Białystok Technical University
Poland

Summary. In the paper the concept of stacks is formalized. As the main result the Theorem of Representation for Stacks is given. Formalization is done according to [13].

MML identifier: `STACKS_1`, version: `7.11.07 4.160.1126`

The papers [6], [15], [14], [2], [4], [7], [16], [8], [9], [10], [5], [1], [17], [11], [19], [21], [20], [3], [18], and [12] provide the terminology and notation for this paper.

1. Introductions

In this paper i is a natural number and x is a set.

Let A be a set and let s_1, s_2 be finite sequences of elements of A. Then $s_1 \frown s_2$ is an element of A^*.

Let A be a set, let i be a natural number, and let s be a finite sequence of elements of A. Then $s_{\restriction i}$ is an element of A^*.

The following two propositions are true:

(1) $\emptyset_{\restriction i} = \emptyset$.

(2) Let D be a non empty set and s be a finite sequence of elements of D. Suppose $s \neq \emptyset$. Then there exists a finite sequence w of elements of D and there exists an element n of D such that $s = \langle n \rangle \frown w$.

The scheme *IndSeqD* deals with a non empty set $\mathcal{A}$ and a unary predicate $\mathcal{P}$, and states that:

For every finite sequence p of elements of $\mathcal{A}$ holds $\mathcal{P}[p]$

provided the following conditions are met:

- $\mathcal{P}[\varepsilon_{\mathcal{A}}]$, and

ISSN 1426–2630(p), 1898-9934(e)

- For every finite sequence p of elements of $\mathcal{A}$ and for every element x of $\mathcal{A}$ such that $\mathcal{P}[p]$ holds $\mathcal{P}[\langle x\rangle \frown p]$.

Let C, D be non empty sets and let R be a binary relation. A function from $C \times D$ into D is said to be a binary operation of C and D being congruence w.r.t. R if:

(Def. 1) For every element x of C and for all elements y_1, y_2 of D such that $\langle y_1, y_2\rangle \in R$ holds $\langle \mathrm{it}(x, y_1), \mathrm{it}(x, y_2)\rangle \in R$.

The scheme *LambdaD2* deals with non empty sets $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ and a binary functor $\mathcal{F}$ yielding an element of $\mathcal{C}$, and states that:

There exists a function M from $\mathcal{A} \times \mathcal{B}$ into $\mathcal{C}$ such that for every element i of $\mathcal{A}$ and for every element j of $\mathcal{B}$ holds $M(i, j) = \mathcal{F}(i, j)$

for all values of the parameters.

Let C, D be non empty sets, let R be an equivalence relation of D, and let b be a function from $C \times D$ into D. Let us assume that b is a binary operation of C and D being congruence w.r.t. R. The functor $b_{/R}$ yielding a function from $C \times \mathrm{Classes}\, R$ into $\mathrm{Classes}\, R$ is defined as follows:

(Def. 2) For all sets x, y, y_1 such that $x \in C$ and $y \in \mathrm{Classes}\, R$ and $y_1 \in y$ holds $b_{/R}(x, y) = [b(x, y_1)]_R$.

Let A, B be non empty sets, let C be a subset of A, let D be a subset of B, let f be a function from A into B, and let g be a function from C into D. Then $f{+\cdot}g$ is a function from A into B.

2. Stack Algebra

We introduce stack systems which are extensions of 2-sorted and are systems
⟨ a carrier, a carrier', empty stacks, a push function, a pop function, a top function ⟩,
where the carrier is a set, the carrier' is a set, the empty stacks constitute a subset of the carrier', the push function is a function from the carrier×the carrier' into the carrier', the pop function is a function from the carrier' into the carrier', and the top function is a function from the carrier' into the carrier.

Let a_1 be a non empty set, let a_2 be a set, let a_3 be a subset of a_2, let a_4 be a function from $a_1 \times a_2$ into a_2, let a_5 be a function from a_2 into a_2, and let a_6 be a function from a_2 into a_1. Observe that stack system$\langle a_1, a_2, a_3, a_4, a_5, a_6\rangle$ is non empty.

Let a_1 be a set, let a_2 be a non empty set, let a_3 be a subset of a_2, let a_4 be a function from $a_1 \times a_2$ into a_2, let a_5 be a function from a_2 into a_2, and let a_6 be a function from a_2 into a_1. One can verify that stack system$\langle a_1, a_2, a_3, a_4, a_5, a_6\rangle$ is non void.

Let us note that there exists a stack system which is non empty, non void, and strict.

Let X be a stack system. A stack of X is an element of the carrier' of X.

Let X be a non empty non void stack system and let s be a stack of X. The predicate $\mathrm{empty}(s)$ is defined by:

(Def. 3) $s \in$ the empty stacks of X.

The functor $\mathrm{pop}\, s$ yields a stack of X and is defined by:

(Def. 4) $\mathrm{pop}\, s =$ (the pop function of X)(s).

The functor $\mathrm{top}\, s$ yields an element of X and is defined by:

(Def. 5) $\mathrm{top}\, s =$ (the top function of X)(s).

Let e be an element of X. The functor $\mathrm{push}(e, s)$ yields a stack of X and is defined by:

(Def. 6) $\mathrm{push}(e, s) =$ (the push function of X)(e,s).

Let A be a non empty set. Standard stack system over A yielding a non empty non void strict stack system is defined by the conditions (Def. 7).

(Def. 7)(i) The carrier of standard stack system over $A = A$,

(ii) the carrier' of standard stack system over $A = A^*$, and

(iii) for every stack s of standard stack system over A holds $\mathrm{empty}(s)$ iff s is empty and for every finite sequence g such that $g = s$ holds if not $\mathrm{empty}(s)$, then $\mathrm{top}\, s = g(1)$ and $\mathrm{pop}\, s = g_{\restriction 1}$ and if $\mathrm{empty}(s)$, then $\mathrm{top}\, s =$ the element of standard stack system over A and $\mathrm{pop}\, s = \emptyset$ and for every element e of standard stack system over A holds $\mathrm{push}(e, s) = \langle e \rangle ^\frown g$.

In the sequel A denotes a non empty set, c denotes an element of standard stack system over A, and m denotes a stack of standard stack system over A.

Let us consider A. Note that every stack of standard stack system over A is relation-like and function-like.

Let us consider A. Observe that every stack of standard stack system over A is finite sequence-like.

We adopt the following convention: X denotes a non empty non void stack system, s, s_1 denote stacks of X, and e, e_1, e_2 denote elements of X.

Let us consider X. We say that X is pop-finite if and only if the condition (Def. 8) is satisfied.

(Def. 8) Let f be a function from $\mathbb{N}$ into the carrier' of X. Then there exists a natural number i and there exists s such that $f(i) = s$ and if not $\mathrm{empty}(s)$, then $f(i + 1) \neq \mathrm{pop}\, s$.

We say that X is push-pop if and only if:

(Def. 9) If not $\mathrm{empty}(s)$, then $s = \mathrm{push}(\mathrm{top}\, s, \mathrm{pop}\, s)$.

We say that X is top-push if and only if:

(Def. 10) $e = \mathrm{top}\,\mathrm{push}(e, s)$.

We say that X is pop-push if and only if:

(Def. 11) $s = \mathrm{pop}\,\mathrm{push}(e, s)$.

We say that X is push-non-empty if and only if:

(Def. 12) not empty(push(e, s)).

Let A be a non empty set. One can verify the following observations:

* standard stack system over A is pop-finite,
* standard stack system over A is push-pop,
* standard stack system over A is top-push,
* standard stack system over A is pop-push, and
* standard stack system over A is push-non-empty.

Let us observe that there exists a non empty non void stack system which is pop-finite, push-pop, top-push, pop-push, push-non-empty, and strict.

A stack algebra is a pop-finite push-pop top-push pop-push push-non-empty non empty non void stack system.

Next we state the proposition

(3) For every non empty non void stack system X such that X is pop-finite there exists a stack s of X such that empty(s).

Let X be a pop-finite non empty non void stack system. Note that the empty stacks of X is non empty.

We now state two propositions:

(4) If X is top-push and pop-push and push(e_1, s_1) = push(e_2, s_2), then $e_1 = e_2$ and $s_1 = s_2$.

(5) If X is push-pop and not empty(s_1) and not empty(s_2) and pop s_1 = pop s_2 and top s_1 = top s_2, then $s_1 = s_2$.

3. Schemes of Induction

Now we present three schemes. The scheme *INDsch* deals with a stack algebra $\mathcal{A}$, a stack $\mathcal{B}$ of $\mathcal{A}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{P}[\mathcal{B}]$

provided the following conditions are satisfied:

- For every stack s of $\mathcal{A}$ such that empty(s) holds $\mathcal{P}[s]$, and
- For every stack s of $\mathcal{A}$ and for every element e of $\mathcal{A}$ such that $\mathcal{P}[s]$ holds $\mathcal{P}[\text{push}(e, s)]$.

The scheme *EXsch* deals with a stack algebra $\mathcal{A}$, a stack $\mathcal{B}$ of $\mathcal{A}$, a non empty set $\mathcal{C}$, an element $\mathcal{D}$ of $\mathcal{C}$, and a binary functor $\mathcal{F}$ yielding an element of $\mathcal{C}$, and states that:

There exists an element a of $\mathcal{C}$ and there exists a function F from the carrier' of $\mathcal{A}$ into $\mathcal{C}$ such that

(i) $a = F(\mathcal{B})$,

(ii) for every stack s_1 of $\mathcal{A}$ such that empty(s_1) holds $F(s_1) = \mathcal{D}$, and

(iii) for every stack s_1 of $\mathcal{A}$ and for every element e of $\mathcal{A}$ holds $F(\mathrm{push}(e, s_1)) = \mathcal{F}(e, F(s_1))$

for all values of the parameters.

The scheme *UNIQsch* deals with a stack algebra $\mathcal{A}$, a stack $\mathcal{B}$ of $\mathcal{A}$, a non empty set $\mathcal{C}$, an element $\mathcal{D}$ of $\mathcal{C}$, and a binary functor $\mathcal{F}$ yielding an element of $\mathcal{C}$, and states that:

Let a_1, a_2 be elements of $\mathcal{C}$. Suppose that

(i) there exists a function F from the carrier' of $\mathcal{A}$ into $\mathcal{C}$ such that $a_1 = F(\mathcal{B})$ and for every stack s_1 of $\mathcal{A}$ such that empty(s_1) holds $F(s_1) = \mathcal{D}$ and for every stack s_1 of $\mathcal{A}$ and for every element e of $\mathcal{A}$ holds $F(\mathrm{push}(e, s_1)) = \mathcal{F}(e, F(s_1))$, and

(ii) there exists a function F from the carrier' of $\mathcal{A}$ into $\mathcal{C}$ such that $a_2 = F(\mathcal{B})$ and for every stack s_1 of $\mathcal{A}$ such that empty(s_1) holds $F(s_1) = \mathcal{D}$ and for every stack s_1 of $\mathcal{A}$ and for every element e of $\mathcal{A}$ holds $F(\mathrm{push}(e, s_1)) = \mathcal{F}(e, F(s_1))$.

Then $a_1 = a_2$

for all values of the parameters.

4. Stack Congruence

We adopt the following rules: X is a stack algebra, s, s_1, s_2, s_3 are stacks of X, and e, e_1, e_2, e_3 are elements of X.

Let us consider X, s. The functor $|s|$ yielding an element of (the carrier of $X)^*$ is defined by the condition (Def. 13).

(Def. 13) There exists a function F from the carrier' of X into (the carrier of $X)^*$ such that $|s| = F(s)$ and for every s_1 such that empty(s_1) holds $F(s_1) = \emptyset$ and for all s_1, e holds $F(\mathrm{push}(e, s_1)) = \langle e \rangle \frown F(s_1)$.

Next we state several propositions:

(6) If empty(s), then $|s| = \emptyset$.

(7) If not empty(s), then $|s| = \langle \mathrm{top}\, s \rangle \frown |\mathrm{pop}\, s|$.

(8) If not empty(s), then $|\mathrm{pop}\, s| = |s|_{\restriction 1}$.

(9) $|\mathrm{push}(e, s)| = \langle e \rangle \frown |s|$.

(10) If not empty(s), then $\mathrm{top}\, s = |s|(1)$.

(11) If $|s| = \emptyset$, then empty(s).

(12) For every stack s of standard stack system over A holds $|s| = s$.

(13) For every element x of (the carrier of $X)^*$ there exists s such that $|s| = x$.

Let us consider X, s_1, s_2. The predicate $s_1 =_G s_2$ is defined as follows:

(Def. 14) $|s_1| = |s_2|$.

Let us notice that the predicate $s_1 =_G s_2$ is reflexive and symmetric.

The following propositions are true:

(14) If $s_1 =_G s_2$ and $s_2 =_G s_3$, then $s_1 =_G s_3$.

(15) If $s_1 =_G s_2$ and empty(s_1), then empty(s_2).

(16) If empty(s_1) and empty(s_2), then $s_1 =_G s_2$.

(17) If $s_1 =_G s_2$, then $\mathrm{push}(e, s_1) =_G \mathrm{push}(e, s_2)$.

(18) If $s_1 =_G s_2$ and not empty(s_1), then $\mathrm{pop}\, s_1 =_G \mathrm{pop}\, s_2$.

(19) If $s_1 =_G s_2$ and not empty(s_1), then $\mathrm{top}\, s_1 = \mathrm{top}\, s_2$.

Let us consider X. We say that X is proper for identity if and only if:

(Def. 15) For all s_1, s_2 such that $s_1 =_G s_2$ holds $s_1 = s_2$.

Let us consider A. Observe that standard stack system over A is proper for identity.

Let us consider X. The functor $==_X$ yields a binary relation on the carrier' of X and is defined as follows:

(Def. 16) $\langle s_1, s_2 \rangle \in ==_X$ iff $s_1 =_G s_2$.

Let us consider X. Observe that $==_X$ is total, symmetric, and transitive.

One can prove the following proposition

(20) If empty(s), then $[s]_{==_X}$ = the empty stacks of X.

Let us consider X, s. The functor coset s yielding a subset of the carrier' of X is defined by the conditions (Def. 17).

(Def. 17)(i) $s \in \mathrm{coset}\, s$,

(ii) for all e, s_1 such that $s_1 \in \mathrm{coset}\, s$ holds $\mathrm{push}(e, s_1) \in \mathrm{coset}\, s$ and if not empty(s_1), then $\mathrm{pop}\, s_1 \in \mathrm{coset}\, s$, and

(iii) for every subset A of the carrier' of X such that $s \in A$ and for all e, s_1 such that $s_1 \in A$ holds $\mathrm{push}(e, s_1) \in A$ and if not empty(s_1), then $\mathrm{pop}\, s_1 \in A$ holds $\mathrm{coset}\, s \subseteq A$.

Next we state three propositions:

(21) If $\mathrm{push}(e, s) \in \mathrm{coset}\, s_1$, then $s \in \mathrm{coset}\, s_1$ and if not empty(s) and $\mathrm{pop}\, s \in \mathrm{coset}\, s_1$, then $s \in \mathrm{coset}\, s_1$.

(22) $s \in \mathrm{coset}\, \mathrm{push}(e, s)$ and if not empty(s), then $s \in \mathrm{coset}\, \mathrm{pop}\, s$.

(23) There exists s_1 such that empty(s_1) and $s_1 \in \mathrm{coset}\, s$.

Let us consider A and let R be a binary relation on A. Note that there exists a reduction sequence w.r.t. R which is A-valued.

Let us consider X. The construction reduction X yielding a binary relation on the carrier' of X is defined as follows:

(Def. 18) $\langle s_1, s_2 \rangle \in$ the construction reduction X iff not empty(s_1) and $s_2 = \mathrm{pop}\, s_1$ or there exists e such that $s_2 = \mathrm{push}(e, s_1)$.

Next we state the proposition

(24) Let R be a binary relation on A and t be a reduction sequence w.r.t. R. Then $t(1) \in A$ if and only if t is A-valued.

The scheme *PathIND* deals with a non empty set $\mathcal{A}$, elements $\mathcal{B}$, $\mathcal{C}$ of $\mathcal{A}$, a binary relation $\mathcal{D}$ on $\mathcal{A}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{P}[\mathcal{C}]$

provided the parameters meet the following conditions:

- $\mathcal{P}[\mathcal{B}]$,
- $\mathcal{D}$ reduces $\mathcal{B}$ to $\mathcal{C}$, and
- For all elements x, y of $\mathcal{A}$ such that $\mathcal{D}$ reduces $\mathcal{B}$ to x and $\langle x, y\rangle \in \mathcal{D}$ and $\mathcal{P}[x]$ holds $\mathcal{P}[y]$.

One can prove the following propositions:

(25) For every reduction sequence t w.r.t. the construction reduction X such that $s = t(1)$ holds $\operatorname{rng} t \subseteq \operatorname{coset} s$.

(26) $\operatorname{coset} s = \{s_1 :$ the construction reduction Xreduces s to $s_1\}$.

Let us consider X, s. The functor $\operatorname{core} s$ yields a stack of X and is defined by the conditions (Def. 19).

(Def. 19)(i) $\operatorname{empty}(\operatorname{core} s)$, and

(ii) there exists a the carrier' of X-valued reduction sequence t w.r.t. the construction reduction X such that $t(1) = s$ and $t(\operatorname{len} t) = \operatorname{core} s$ and for every i such that $1 \leq i < \operatorname{len} t$ holds not $\operatorname{empty}(t_i)$ and $t_{i+1} = \operatorname{pop}(t_i)$.

The following propositions are true:

(27) If $\operatorname{empty}(s)$, then $\operatorname{core} s = s$.

(28) $\operatorname{core} \operatorname{push}(e, s) = \operatorname{core} s$.

(29) If not $\operatorname{empty}(s)$, then $\operatorname{core} \operatorname{pop} s = \operatorname{core} s$.

(30) $\operatorname{core} s \in \operatorname{coset} s$.

(31) For every element x of (the carrier of X)* there exists s_1 such that $|s_1| = x$ and $s_1 \in \operatorname{coset} s$.

(32) If $s_1 \in \operatorname{coset} s$, then $\operatorname{core} s_1 = \operatorname{core} s$.

(33) If s_1, $s_2 \in \operatorname{coset} s$ and $|s_1| = |s_2|$, then $s_1 = s_2$.

(34) There exists s such that $\operatorname{coset} s_1 \cap [s_2]_{==_X} = \{s\}$.

5. Quotient Stack System

Let us consider X. The functor $X_{/==}$ yields a strict stack system and is defined by the conditions (Def. 20).

(Def. 20)(i) The carrier of $X_{/==}$ = the carrier of X,

(ii) the carrier' of $X_{/==} = \operatorname{Classes} ==_X$,

(iii) the empty stacks of $X_{/==} = \{$the empty stacks of $X\}$,

(iv) the push function of $X_{/==}$ = (the push function of $X)_{/==_X}$,

(v) the pop function of $X_{/==}$ =
((the pop function of X)$+\cdot\operatorname{id}_{\text{the empty stacks of } X})_{/==_X}$, and

(vi) for every choice function f of Classes $==_X$ holds the top function of $X_{/==}$ = (the top function of X) $\cdot f$ +$\cdot$ (the empty stacks of X, the element of the carrier of X).

Let us consider X. One can verify that $X_{/==}$ is non empty and non void.

The following propositions are true:

(35) For every stack S of $X_{/==}$ there exists s such that $S = [s]_{==_X}$.

(36) $[s]_{==_X}$ is a stack of $X_{/==}$.

(37) For every stack S of $X_{/==}$ such that $S = [s]_{==_X}$ holds empty(s) iff empty(S).

(38) For every stack S of $X_{/==}$ holds empty(S) iff S = the empty stacks of X.

(39) For every stack S of $X_{/==}$ and for every element E of $X_{/==}$ such that $S = [s]_{==_X}$ and $E = e$ holds push$(e, s) \in$ push(E, S) and $[\text{push}(e, s)]_{==_X} = \text{push}(E, S)$.

(40) For every stack S of $X_{/==}$ such that $S = [s]_{==_X}$ and not empty(s) holds $\text{pop}\, s \in \text{pop}\, S$ and $[\text{pop}\, s]_{==_X} = \text{pop}\, S$.

(41) For every stack S of $X_{/==}$ such that $S = [s]_{==_X}$ and not empty(s) holds $\text{top}\, S = \text{top}\, s$.

Let us consider X. One can verify the following observations:

* $X_{/==}$ is pop-finite,
* $X_{/==}$ is push-pop,
* $X_{/==}$ is top-push,
* $X_{/==}$ is pop-push, and
* $X_{/==}$ is push-non-empty.

Next we state the proposition

(42) For every stack S of $X_{/==}$ such that $S = [s]_{==_X}$ holds $|S| = |s|$.

Let us consider X. Note that $X_{/==}$ is proper for identity.

Let us note that there exists a stack algebra which is proper for identity.

6. Representation Theorem for Stacks

Let X_1, X_2 be stack algebras and let F, G be functions. We say that F and G form isomorphism between X_1 and X_2 if and only if the conditions (Def. 21) are satisfied.

(Def. 21) $\text{dom}\, F$ = the carrier of X_1 and $\text{rng}\, F$ = the carrier of X_2 and F is one-to-one and $\text{dom}\, G$ = the carrier' of X_1 and $\text{rng}\, G$ = the carrier' of X_2 and G is one-to-one and for every stack s_1 of X_1 and for every stack s_2 of X_2 such that $s_2 = G(s_1)$ holds empty(s_1) iff empty(s_2) and if not empty(s_1), then $\text{pop}\, s_2 = G(\text{pop}\, s_1)$ and $\text{top}\, s_2 = F(\text{top}\, s_1)$ and for every element

e_1 of X_1 and for every element e_2 of X_2 such that $e_2 = F(e_1)$ holds $\mathrm{push}(e_2, s_2) = G(\mathrm{push}(e_1, s_1))$.

We use the following convention: X_1, X_2, X_3 are stack algebras and F, F_1, F_2, G, G_1, G_2 are functions.

The following propositions are true:

(43) $\mathrm{id}_{\text{the carrier of } X}$ and $\mathrm{id}_{\text{the carrier' of } X}$ form isomorphism between X and X.

(44) If F and G form isomorphism between X_1 and X_2, then F^{-1} and G^{-1} form isomorphism between X_2 and X_1.

(45) Suppose F_1 and G_1 form isomorphism between X_1 and X_2 and F_2 and G_2 form isomorphism between X_2 and X_3. Then $F_2 \cdot F_1$ and $G_2 \cdot G_1$ form isomorphism between X_1 and X_3.

(46) Suppose F and G form isomorphism between X_1 and X_2. Let s_1 be a stack of X_1 and s_2 be a stack of X_2. If $s_2 = G(s_1)$, then $|s_2| = F \cdot |s_1|$.

Let X_1, X_2 be stack algebras. We say that X_1 and X_2 are isomorphic if and only if:

(Def. 22) There exist functions F, G such that F and G form isomorphism between X_1 and X_2.

Let us notice that the predicate X_1 and X_2 are isomorphic is reflexive and symmetric.

We now state four propositions:

(47) If X_1 and X_2 are isomorphic and X_2 and X_3 are isomorphic, then X_1 and X_3 are isomorphic.

(48) If X_1 and X_2 are isomorphic and X_1 is proper for identity, then X_2 is proper for identity.

(49) Let X be a proper for identity stack algebra. Then there exists G such that

(i) for every stack s of X holds $G(s) = |s|$, and

(ii) $\mathrm{id}_{\text{the carrier of } X}$ and G form isomorphism between X and standard stack system over the carrier of X.

(50) Let X be a proper for identity stack algebra. Then X and standard stack system over the carrier of X are isomorphic.

References

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

[2] Grzegorz Bancerek. Filters – part II. Quotient lattices modulo filters and direct product of two lattices. *Formalized Mathematics*, 2(**3**):433–438, 1991.

[3] Grzegorz Bancerek. Reduction relations. *Formalized Mathematics*, 5(**4**):469–478, 1996.

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

[5] Grzegorz Bancerek and Andrzej Trybulec. Miscellaneous facts about functions. *Formalized Mathematics*, 5(**4**):485–492, 1996.

[6] Czesław Byliński. Binary operations. *Formalized Mathematics*, 1(**1**):175–180, 1990.

[7] Czesław Byliński. Finite sequences and tuples of elements of a non-empty sets. *Formalized Mathematics*, 1(**3**):529–536, 1990.

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

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

[10] Czesław Byliński. The modification of a function by a function and the iteration of the composition of a function. *Formalized Mathematics*, 1(**3**):521–527, 1990.

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

[12] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.

[13] Grażyna Mirkowska and Andrzej Salwicki. *Algorithmic Logic*. PWN-Polish Scientific Publisher, 1987.

[14] Konrad Raczkowski and Paweł Sadowski. Equivalence relations and classes of abstraction. *Formalized Mathematics*, 1(**3**):441–444, 1990.

[15] Andrzej Trybulec. Domains and their Cartesian products. *Formalized Mathematics*, 1(**1**):115–122, 1990.

[16] Wojciech A. Trybulec. Non-contiguous substrings and one-to-one finite sequences. *Formalized Mathematics*, 1(**3**):569–573, 1990.

[17] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[18] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

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

[20] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

[21] Edmund Woronowicz and Anna Zalewska. Properties of binary relations. *Formalized Mathematics*, 1(**1**):85–89, 1990.

Received February 22, 2011