FORMALIZED MATHEMATICS
Volume 12, Number 1, 2004
University of Białystok

A Tree of Execution of a Macroinstruction[1]

Artur Korniłowicz
University of Białystok

Summary. A tree of execution of a macroinstruction is defined. It is a tree decorated by the instruction locations of a computer. Successors of each vertex are determined by the set of all possible values of the instruction counter after execution of the instruction placed in the location indicated by given vertex.

MML Identifier: AMISTD_3.

The articles [22], [14], [25], [15], [1], [20], [3], [4], [16], [26], [11], [13], [12], [5], [6], [21], [9], [8], [10], [2], [7], [18], [23], [19], [24], and [17] provide the notation and terminology for this paper.

For simplicity, we adopt the following convention: x, y, X are sets, m, n are natural numbers, O is an ordinal number, and R, S are binary relations.

Let D be a set, let f be a partial function from D to $\mathbb{N}$, and let n be a set. One can verify that $f(n)$ is natural.

Let R be an empty binary relation and let X be a set. Observe that $R{\restriction}X$ is empty.

One can prove the following two propositions:

(1) If $\operatorname{dom} R = \{x\}$ and $\operatorname{rng} R = \{y\}$, then $R = x \dot{\longmapsto} y$.

(2) $\operatorname{field}\{\langle x, x\rangle\} = \{x\}$.

Let X be an infinite set and let a be a set. One can verify that $X \longmapsto a$ is infinite.

One can check that there exists a function which is infinite.

Let R be a finite binary relation. One can verify that $\operatorname{field} R$ is finite.

The following proposition is true

(3) If $\operatorname{field} R$ is finite, then R is finite.

[1]The paper was written during author's post-doctoral fellowship granted by Shinshu University, Japan.

ISSN 1426–2630

Let R be an infinite binary relation. Note that field R is infinite.

One can prove the following proposition

(4) If dom R is finite and rng R is finite, then R is finite.

Let us observe that $\subseteq_{\emptyset}$ is empty.

Let X be a non empty set. One can verify that $\subseteq_X$ is non empty.

Next we state two propositions:

(5) $\subseteq_{\{x\}} = \{\langle x, x\rangle\}$.

(6) $\subseteq_X \subseteq [: X, X :]$.

Let X be a finite set. Note that $\subseteq_X$ is finite.

One can prove the following proposition

(7) If $\subseteq_X$ is finite, then X is finite.

Let X be an infinite set. One can verify that $\subseteq_X$ is infinite.

The following propositions are true:

(8) If R and S are isomorphic and R is well-ordering, then S is well-ordering.

(9) If R and S are isomorphic and R is finite, then S is finite.

(10) $x \dot{\longmapsto} y$ is an isomorphism between $\{\langle x, x\rangle\}$ and $\{\langle y, y\rangle\}$.

(11) $\{\langle x, x\rangle\}$ and $\{\langle y, y\rangle\}$ are isomorphic.

One can verify that $\overline{\overline{\emptyset}}$ is empty.

The following propositions are true:

(12) $\overline{\overline{\subseteq_O}} = O$.

(13) For every finite set X such that $X \subseteq O$ holds $\overline{\overline{\subseteq_X}} = \operatorname{card} X$.

(14) If $\{x\} \subseteq O$, then $\overline{\overline{\subseteq_{\{x\}}}} = 1$.

(15) If $\{x\} \subseteq O$, then the canonical isomorphism between $\subseteq_{\overline{\overline{\subseteq_{\{x\}}}}}$ and $\subseteq_{\{x\}} = 0 \dot{\longmapsto} x$.

Let O be an ordinal number, let X be a subset of O, and let n be a set. One can check that (the canonical isomorphism between $\subseteq_{\overline{\overline{\subseteq_X}}}$ and $\subseteq_X)(n)$ is ordinal.

Let X be a natural-membered set and let n be a set. Note that (the canonical isomorphism between $\subseteq_{\overline{\overline{\subseteq_X}}}$ and $\subseteq_X)(n)$ is natural.

Next we state three propositions:

(16) If $n \mapsto x = m \mapsto x$, then $n = m$.

(17) For every tree T and for every element t of T holds $t \restriction \operatorname{Seg} n \in T$.

(18) For all trees T_1, T_2 such that for every natural number n holds T_1-level$(n) = T_2$-level(n) holds $T_1 = T_2$.

The functor TrivialInfiniteTree is defined by:

(Def. 1) TrivialInfiniteTree $= \{k \mapsto 0 : k$ ranges over natural numbers$\}$.

One can check that TrivialInfiniteTree is non empty and tree-like.

We now state the proposition

(19) $\mathbb{N} \approx$ TrivialInfiniteTree .

Let us note that TrivialInfiniteTree is infinite.

The following proposition is true

(20) For every natural number n holds TrivialInfiniteTree-level$(n) = \{n \mapsto 0\}$.

For simplicity, we adopt the following convention: N denotes a set with non empty elements, S denotes a standard IC-Ins-separated definite non empty non void AMI over N, L, l_1 denote instruction-locations of S, J denotes an instruction of S, and F denotes a subset of the instruction locations of S.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let F be a finite partial state of S. Let us assume that F is non empty and F is programmed. The functor FirstLoc(F) yields an instruction-location of S and is defined by the condition (Def. 2).

(Def. 2) There exists a non empty subset M of $\mathbb{N}$ such that $M = \{\mathrm{locnum}(l); l$ ranges over elements of the instruction locations of S: $l \in \mathrm{dom}\, F\}$ and FirstLoc$(F) = \mathrm{il}_S(\min M)$.

One can prove the following four propositions:

(21) For every non empty programmed finite partial state F of S holds FirstLoc$(F) \in \mathrm{dom}\, F$.

(22) For all non empty programmed finite partial states F, G of S such that $F \subseteq G$ holds FirstLoc$(G) \leqslant$ FirstLoc(F).

(23) For every non empty programmed finite partial state F of S such that $l_1 \in \mathrm{dom}\, F$ holds FirstLoc$(F) \leqslant l_1$.

(24) For every lower non empty programmed finite partial state F of S holds FirstLoc$(F) = \mathrm{il}_S(0)$.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let F be a subset of the instruction locations of S. The functor LocNums(F) yields a subset of $\mathbb{N}$ and is defined by:

(Def. 3) LocNums$(F) = \{\mathrm{locnum}(l); l$ ranges over instruction-locations of S: $l \in F\}$.

We now state the proposition

(25) $\mathrm{locnum}(l_1) \in$ LocNums(F) iff $l_1 \in F$.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let F be an empty subset of the instruction locations of S. Observe that LocNums(F) is empty.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let F be a non empty subset of the instruction locations of S. Observe that LocNums(F) is non empty.

We now state several propositions:

(26) If $F = \{\mathrm{il}_S(n)\}$, then LocNums$(F) = \{n\}$.

(27) $F \approx \mathrm{LocNums}(F)$.

(28) $\overline{\overline{F}} \subseteq \overline{\subseteq_{\mathrm{LocNums}(F)}}$.

(29) If S is realistic and J is halting, then $\mathrm{LocNums}(\mathrm{NIC}(J, L)) = \{\mathrm{locnum}(L)\}$.

(30) If S is realistic and J is sequential, then $\mathrm{LocNums}(\mathrm{NIC}(J, L)) = \{\mathrm{locnum}(\mathrm{NextLoc}\, L)\}$.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let M be a subset of the instruction locations of S. The functor $\mathrm{LocSeq}(M)$ yielding a transfinite sequence of elements of the instruction locations of S is defined as follows:

(Def. 4) $\mathrm{dom}\,\mathrm{LocSeq}(M) = \overline{\overline{M}}$ and for every set m such that $m \in \overline{\overline{M}}$ holds $(\mathrm{LocSeq}(M))(m) = \mathrm{il}_S(($the canonical isomorphism between $\overline{\subseteq_{\subseteq_{\mathrm{LocNums}(M)}}}$ and $\subseteq_{\mathrm{LocNums}(M)})(m))$.

One can prove the following proposition

(31) If $F = \{\mathrm{il}_S(n)\}$, then $\mathrm{LocSeq}(F) = 0 \dot{\longmapsto} \mathrm{il}_S(n)$.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let M be a subset of the instruction locations of S. Note that $\mathrm{LocSeq}(M)$ is one-to-one.

Let N be a set with non empty elements, let S be a standard IC-Ins-separated definite non empty non void AMI over N, and let M be a finite partial state of S. The functor $\mathrm{ExecTree}(M)$ yields a tree decorated with elements of the instruction locations of S and is defined by the conditions (Def. 5).

(Def. 5)(i) $(\mathrm{ExecTree}(M))(\emptyset) = \mathrm{FirstLoc}(M)$, and

(ii) for every element t of $\mathrm{dom}\,\mathrm{ExecTree}(M)$ holds $\mathrm{succ}\, t = \{t ^\frown \langle k \rangle; k$ ranges over natural numbers: $k \in \overline{\overline{\mathrm{NIC}(\pi_{(\mathrm{ExecTree}(M))(t)} M, (\mathrm{ExecTree}(M))(t))}}\}$ and for every natural number m such that $m \in \overline{\overline{\mathrm{NIC}(\pi_{(\mathrm{ExecTree}(M))(t)} M, (\mathrm{ExecTree}(M))(t))}}$ holds $(\mathrm{ExecTree}(M))(t ^\frown \langle m \rangle) = (\mathrm{LocSeq}(\mathrm{NIC}(\pi_{(\mathrm{ExecTree}(M))(t)} M, (\mathrm{ExecTree}(M))(t))))(m)$.

One can prove the following proposition

(32) For every standard halting realistic IC-Ins-separated definite non empty non void AMI S over N holds $\mathrm{ExecTree}(\mathrm{Stop}\, S) = \mathrm{TrivialInfiniteTree} \longmapsto \mathrm{il}_S(0)$.

Acknowledgments

The author wishes to thank Andrzej Trybulec and Grzegorz Bancerek for their very useful comments during writing this article.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. Introduction to trees. *Formalized Mathematics*, 1(**2**):421–427, 1990.
[3] Grzegorz Bancerek. The ordinal numbers. *Formalized Mathematics*, 1(**1**):91–96, 1990.
[4] Grzegorz Bancerek. Sequences of ordinal numbers. *Formalized Mathematics*, 1(**2**):281–290, 1990.
[5] Grzegorz Bancerek. The well ordering relations. *Formalized Mathematics*, 1(**1**):123–129, 1990.
[6] Grzegorz Bancerek. Zermelo theorem and axiom of choice. *Formalized Mathematics*, 1(**2**):265–267, 1990.
[7] Grzegorz Bancerek. König's lemma. *Formalized Mathematics*, 2(**3**):397–402, 1991.
[8] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.
[9] Czesław Byliński. A classical first order language. *Formalized Mathematics*, 1(**4**):669–676, 1990.
[10] Czesław Byliński. Finite sequences and tuples of elements of a non-empty sets. *Formalized Mathematics*, 1(**3**):529–536, 1990.
[11] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.
[12] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.
[13] Czesław Byliński. Partial functions. *Formalized Mathematics*, 1(**2**):357–367, 1990.
[14] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.
[15] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.
[16] Agata Darmochwał and Andrzej Trybulec. Similarity of formulae. *Formalized Mathematics*, 2(**5**):635–642, 1991.
[17] Artur Korniłowicz. On the composition of macro instructions of standard computers. *Formalized Mathematics*, 9(**2**):303–316, 2001.
[18] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.
[19] Yasushi Tanaka. On the decomposition of the states of SCM. *Formalized Mathematics*, 5(**1**):1–8, 1996.
[20] Andrzej Trybulec. Subsets of complex numbers. *To appear in Formalized Mathematics.*
[21] Andrzej Trybulec. Binary operations applied to functions. *Formalized Mathematics*, 1(**2**):329–334, 1990.
[22] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.
[23] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.
[24] Andrzej Trybulec, Piotr Rudnicki, and Artur Korniłowicz. Standard ordering of instruction locations. *Formalized Mathematics*, 9(**2**):291–301, 2001.
[25] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.
[26] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

Received December 10, 2003