韓國數學教育學會誌<數學教育>
1982. 12. Vol. XXI No. 1

Journal of the Society of Mathematical Education
Dec. 1982. Vol. XXI No. 1

A Proof of the Rule of Consequence in Inference (Programming Language)

By Myoung Lyoul Kim
Jeonbug National University. Jeonju, Korea

1. Introduction

Recent years have witnessed major advances both in the semantics of programming language, and in the methology of program design and verification.

The rule of consequence comes from the soundness of inference which is a part of proof theory on the basis of the concept and semantics.

Addition to this, rule of composition and rule of conditionals consist proof rules.

We introduce three classes of constructs:

Iexp (for integer expressions) with typical elementl s, $t, \cdots$,

Bexp (for boolean expressions) with typical elements $b, \cdots$, and

Stat (for statements) with typical elements $S, \cdots$.

And we have two classes of semantic entities:

Ivar (for integer variables) with typical elements x, y, z, $u, \cdots$, and

Icon (for integer constants) with typical elements m, $n, \cdots$.

V is used to denote the set of integers, with typical elements $\alpha, \cdots$.

B is used to denote the set of truth-values $\{tt, ff\}$, with typical elements $\beta, \cdots$.

$\sum$ is used to denote the set of states, *i.e.*, of all functions from *Ivar* to V, and has typical elements σ, $\cdots$.

We define two functions as follows.

$T : Assn \rightarrow (\sum \rightarrow B)$

$F : Form \rightarrow (\sum \rightarrow B)$

2. Definitions

(1) An *inference* is a construct of the form $\frac{f_1, \cdots, f_n}{f}$, with $f_i, i=1, \cdots, n$, and $f \in Form$.

(2) An inference $\frac{f_1, \cdots, f_n}{f}$ is called *sound*, written as

$$\left| \frac{f_1, \ \cdots, \ f_n}{f} \right|$$

whenever validity of f_i, $i=1, \cdots, n$, implies validity of f.

3. Theorem (rule of consequence)

$$\frac{(p \supset p_1),\ \{p_1\}\ S\{q_1\},\ (q_1 \supset q)}{\{p\}S\{q\}}$$

Proof. Assume that for all σ, $T(p)(\sigma) \Rightarrow T(p_1)(\sigma)$
and, for all σ, $T(p_1)(\sigma) \Rightarrow T(q_1)(M(S)(\sigma))$,
and, for all σ, $T(q_1)(\sigma) \Rightarrow T(q)(\sigma)$.
Now, we take $\sigma' = M(S)(\sigma)$.
Then $T(p)(\sigma') \Rightarrow T(p_1)(\sigma')$,
$T(p_1)(\sigma') \Rightarrow T(q_1)(\sigma')$
and, $T(q_1)(\sigma') \Rightarrow T(q)(\sigma')$,
so that $T(p)(\sigma') \Rightarrow T(q)(\sigma')$.
Hence $T(p)(\sigma) \Rightarrow T(q)(M(S)(\sigma))$.

References

〔1〕 Jaco de Bakker: *Mathematical Theory of Program Correctness;* Prentice-Hall International, Inc., London, 1980, pp. 1-68.

〔2〕 Zahar Manna: *Mathematical Theory of Computation;* McGraw-Hill Kogakusha, Ltd., 1974, pp. 77-160.

〔3〕 朴漢植：數學論理學；서울, 敎學社, 1982, pp. 177-204.