Boolean and Arithmetic Language Evaluation

Ashish Kosana

2 Boolean Arithmetic Language

2.a) Validity of the Term $1 + \text{true}$

The term $1 + \text{true}$ is not a valid term of the boolean and arithmetic language. In this language, addition is defined only for integer terms. Since true is a boolean value, the expression is invalid.

2.b) Small-Step Rules for Term Addition

There are three small-step rules for term addition in the boolean and arithmetic language:

1. Addition of Two Numbers (AddNum):

$$n_1 + n_2 \xrightarrow{\text{AddNum}} n_1 + n_2 = n$$

2. Evaluating the Left Operand (EvalLeft):

$$t_1 + t_2 \xrightarrow{\text{EvalLeft}} t_1' + t_2 \quad \text{if } t_1 \rightarrow t_1'$$

3. Evaluating the Right Operand (EvalRight):

$$n + t_2 \xrightarrow{\text{EvalRight}} n + t_2' \quad \text{if } t_2 \rightarrow t_2'$$

2.c) Evaluating the Term $1 + 2 + 3$

To evaluate the term $1 + 2 + 3$, we follow these steps:

1. Start with the term:
$$1 + 2 + 3$$

2. Apply the EvalLeft rule:
$$1 + 2 + 3 \xrightarrow{\text{EvalLeft}} (1 + 2) + 3$$

3. Evaluate $1 + 2$ using AddNum:
$$(1 + 2) + 3 \xrightarrow{\text{AddNum}} 3 + 3$$

4. Finally, apply AddNum again:
$$3 + 3 \xrightarrow{\text{AddNum}} 6$$

Putting it all together, the complete evaluation sequence is:

$$1 + 2 + 3 \xrightarrow{\text{EvalLeft}} (1 + 2) + 3 \xrightarrow{\text{AddNum}} 3 + 3 \xrightarrow{\text{AddNum}} 6$$