Abstract

Typing Parsing Expression Grammar with Regular Expression Types

Daisuke Yamaguchi[1,a) Kimio Kuramitsu[1]

Presented: November 2, 2017

Using a parser generator, the developers could get a practical parser from a language specification with a formal grammar such a LR(k), LL(k), GLR or parsing expression grammar. However, it is hard to determine the data structure of parsing results from a generated parser statically. Even the developers pay close attention to implement an evaluation function for the parsing results, it is difficult to avoid a run-time error caused by manipulations that do not conform to the data structure of the parsing result. In this paper, we introduce a method of typing parsing expression grammar with regular expression types. Regular expression types, which a parsing expression grammar is typed with, represent a set of abstract syntax trees and describe their structure(i.e. the number of child nodes, the name of the node and the order). We defined typing rules and proved the property of subject reduction. We can describe the structure of abstract syntax trees, which are parsing results of generated parser, with types and verify it with static type checking.

[1] Graduate School of Electronic and Computer Engineering, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan

[a)] yamaguchi-daisuke-bf@ynu.jp