

Parsing Mixfix Operators

Bc. Michal Gabonay (xgabon00@stud.fit.vutbr.cz)

Bc. Martin Kocour (xkocou08@stud.fit.vutbr.cz)

Operators are one of the most important things in programs. Operators can be prefix, infix, postfix or closed. The term mixfix actually refers to something more flexible than just that. The idea of mixfix operators is, that it create an easy way to users define their own operators in a more flexible way than just a single prefix or infix word. In the mixfix viewpoint, many syntactic constructs might be seen as operators that can have precedence in relation to others.

A Grammar of language, which consists of arithmetic and logical expression, should include such precedence rules in its production rules. Writing such production rules into Backus–Naur form may get the grammar quite complex. That is why mixfix parsers add a new abstraction layer for operator precedence. They describe operators and their precedens as a directed acyclic graph, where nodes are groups of operators with same precedence and edges are precedences. Parser combinators are used to implement such graph inside language parser.