

Assignment Series 3

Syntactic Analysis

Consider the following context-free grammar of expressions constructed from identifiers using binary addition, unary negation, post-decrement and post-increment:

<i>Expr</i>	\Rightarrow	Id
		<i>Expr</i> + <i>Expr</i>
		- <i>Expr</i>
		<i>Expr</i> --
		<i>Expr</i> ++

Assignment 11: Precedence and Associativity

Rewrite the above grammar such that it properly expresses precedence and associativity according to the C standard: http://en.wikipedia.org/wiki/Operators_in_C_and_C++

Assignment 12: Left- and Right-recursive Grammars

Convert the (left-recursive) grammar developed for Assignment 11 into a right-recursive grammar.

Assignment 13: Predictive Grammars

Convert the right-recursive grammar of Assignment 12 into a start-separated, predictive grammar.

Assignment 14: Recursive-descent Parsing

Derive pseudo code for a top-down recursive-descent parser from the start-separated, predictive grammar of Assignment 13.