

Compilers Course Project 2022 - Ossi Lehtonen

The Mini-PL token patterns as regular expressions

Assignment = \:|=

Operators = \+|-|*|/|<|>|&|!

Parentheses = \(|\)

String = \"[^\"]*\"

Int = \d*

Idents_and_keywords = [A-Za-z]+(_|\\d)*[A-Za-z]*

Token = String | Int | Idents_and_keywords | \; | \. | : | Assignment | Operators | parentheses

A modified context-free grammar suitable for recursive-descent parsing (eliminating any LL(1) violations); modifications must not affect the language that is accepted.

Specify abstract syntax trees (AST), i.e., the internal representation for Mini-PL programs; you can use UML diagrams or alternatively give a syntax-based definition of the abstract syntax.

Error handling approach and solutions used in your Mini-PL implementation (in its scanner, parser, semantic analyzer, and interpreter).

Work Hours

Add work hours here when project is ready