Notes on Formal Compiler Construction with the π Framework

Christiano Braga

Instituto de Computação, Universidade Federal Fluminense, Niterói, Brazil

August 14, 2018



Compiler architecture

source	\xrightarrow{lexer}	tokens	$\xrightarrow{\textit{parser}}$	concrete	$\xrightarrow{AST \ transformer}$	abstract	$\xrightarrow{type \ checker}$	abstract	$\xrightarrow{code\ generator}$	machine	$\xrightarrow{optimizer}$	optimized
code				syntax		syntax		syntax		code		machine
				tree		tree		tree				code

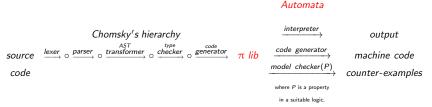


Compiler architecture and formal languages

	Regular Grammar		ContextFree Grammar	ContextFree Grammar			ContextSensitive Grammar		Turing Machine		Turing Machine	
source	\xrightarrow{lexer}	tokens	parser >	concrete	$\xrightarrow{AST \ transformer}$	abstract	type checker	abstract	$\xrightarrow{code\ generator}$	machine	$\xrightarrow{optimizer}$	optimized
code				syntax		syntax		syntax		code		machine
				tree		tree		tree				code



Compiler architecture with the π Framework



- π lib defines a set of constructions common to many programming languages.
- π lib constructions have a formal automata-based semantics in π automata.
- One may execute (or validate) a program in a given language by running its associated π lib program.
- http://github.com/ChristianoBraga/BPLC
- Notes on Formal Compiler Construction with the π Framework: https://github.com/ChristianoBraga/BPLC/blob/master/notes/notes.pdf

