Today
Lab Questions
Concrete & Abstract Syntax

Grammars

Language is a set of strings, sentence inductively-defined

\[ L_1 = \{ \text{"hi"} \} \]
Grammars is a way to describe languages (BNF is a notation for grammars) (Backus-Naur Form)

Context-Free Grammars describe context-free languages

Alphabet - is possible characters

"atomic things"

"terminals in a grammar"

tokens

lexical analysis

raw characters → tokens

\[
\text{lexing} \quad 42 + 1
\]

syntactic analysis = parsing

seq tokens → check in language / not

→ tree (concrete vs. abstract syntax)
\[ e ::= n \mid e_1 + e_2 \]

\[ \text{number token} \]

\[ 42 + 1 \quad 1 + 2 + 3 \]

\[ e \rightarrow e + e \rightarrow n(42) + e \rightarrow n(42) + n(1) \]

*Derivation*

**Ambiguity** = two different parse trees

**Grammar** for a string

**Judgments**

inductively-defined relations

Grammar describes an inductively-defined set

*Stuff to be set of terms described by the grammar above*
\[
\begin{align*}
\text{Inference Rules} \\
\frac{\text{Premises}}{\text{Conclusion}} \\
\frac{\text{Premises}}{\text{Conclusion}} \\
\end{align*}
\]
Lab 2

JavaScript  def  Node

Scala  

JavaScript  x86