CFG3: Push-down Automata

Context-Free Grammars

CMPT 379: Compilers Instructor: Anoop Sarkar anoopsarkar.github.io/compilers-class

Context-free languages and Pushdown Automata

- Recall that for each regular language there was an equivalent finitestate automaton
- The FSA was used as a recognizer of the regular language
- For each context-free language there is also an automaton that recognizes it: called a **pushdown automaton (pda)**

Pushdown Automata



Pushdown Automata



Pushdown Automata

- PDA has
 - an alphabet (terminals),
 - stack symbols (non-terminals and terminals),
 - a finite-state automaton,
 - stack



Two actions

Reduce

Pop symbols from

top of stack and

push a new symbol

Shift

Push symbol

on top of stack

Shift Reduce Parsing

Summary of CFG[123]

- CFGs can be used describe structure of programming languages
- Derivations correspond to parse trees
- Parse trees represent structure of input programs
- There are ambiguous (non-deterministic) CFGs
- Some forms of ambiguity can be fixed by changing the grammar
- Context-free languages can be recognized using Pushdown Automata

Extra Slides

CF Languages

 $L_4 = \{wcw^R \mid w \in (a|b)*\}$ $S \rightarrow aSa \mid bSb \mid c$ $L_5 = \{a^n b^m c^m d^n \mid n \ge 1, m \ge 1\}$ $S \rightarrow aSd \mid aAd$ $A \rightarrow bAc \mid bc$

Non-CF Languages

$$L_{1} = \{wcw \mid w \in (a|b)*\}$$
$$L_{2} = \{a^{n}b^{m}c^{n}d^{m} \mid n \ge 1, m \ge 1\}$$
$$L_{3} = \{a^{n}b^{n}c^{n} \mid n \ge 0\}$$