

Lecture 13+14

Context-Free Languages

CS 241: Foundations of Sequential Programs
Fall 2009

Troy Vasiga et al
University of Waterloo

Non-regular languages

Give a DFA/NFA/ ϵ -NFA/RE over $\Sigma = \{a, b\}$ for

$$L = \{w: \text{numbers of a's in } w = \text{number of b's in } w\}$$

CFLs

- ▶ L is not a regular language, but it is context-free
- ▶ Note that a compiler needs to do this and then some: balanced parenthesis
- ▶ Context-free languages are built from:
 - ▶ finite sets
 - ▶ concatenation
 - ▶ union
 - ▶ recursion
- ▶ recognizers for regular languages use a finite amount of memory
- ▶ recognizers for context-free languages use a finite amount of memory plus one (unbounded) stack

Context-Free Grammars

A way to specify a regular language.

A CFG Example

$$S \rightarrow aSb$$

$$S \rightarrow D$$

$$D \rightarrow cD$$

$$D \rightarrow \epsilon$$

Discussion of example

- ▶ G:
- ▶ $L(G)$:
- ▶ a word:
- ▶ a derivation:
- ▶ alternation and concatenation
- ▶ recursion vs. repetition

Formal Definition

A context-free grammar (CFG) consists of

- ▶ N
- ▶ T
- ▶ P
- ▶ S

From the example, what is N, T, P, S?

Example: balanced parentheses

Example words:

CFG:

Binary expressions

Word: 10+1

CFG:

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.

Derivations

We say that $\alpha A \beta$ *directly derives* $\alpha \gamma \beta$ if there exists a production $A \rightarrow \gamma$.
Also called a derivation step.

We say that $\alpha A \beta$ *derives* $\alpha \gamma \beta$ if

G derives $w \in T^*$ if

$L(G) =$

L is context-free if

Left-most and right-most derivations

Derivations as Proofs

Parse Trees

Example:

Discussion:

Meaning of a parse tree

Problems that grammars can encounter: Ambiguity

A real-world example of ambiguity:

Problems that grammars can encounter: Ambiguity

Consider the binary expression grammar, and $1-10+11$

Problems that grammars can encounter: Ambiguity

Terminology:

Fixing ambiguity

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.

Ambiguity in programming languages

- ▶ Some programming languages have ambiguous grammars
- ▶ Pascal has a “dangling else”
- ▶ “Fixed” by way of a footnote

Associativity

$$1-10+11$$

Fixing associativity problems

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.

Precedence Problems

Adding multiplication to the grammar.

Consider $1*10+11$.

Consider $1+10*11$

Fixing precedence problems

Reg. Lang vs. CFL

Summary