

CompSci 162
Spring 2023 Lecture 8:
Introduction to Context-Free
Grammars

CompSci 162

Spring 2023 Lecture 8:

Introduction to Context-Free Grammars



matt 4:12 PM

man i cannot overstate how happy these pumping lemma proofs are making me, especially given the short amount of time since the first PL lecture. so proud 🙄

Generating a String/Sentence

Start symbol

Describe strings from $a(a^* \cup b^*)b$

Derivation
of $aaab$

$S \rightarrow aMb \leftarrow \text{terminals}$

$\left[\begin{array}{l} M \rightarrow A \\ M \rightarrow B \end{array} \right] \text{Rules}$

$A \rightarrow \epsilon$

$\underline{A} \rightarrow aA$

$B \rightarrow \epsilon$

$B \rightarrow bB$

S

aMb

aAb

$aaAb$

$aaaAb$

$aaab$

► Example: generate the string aaab

Defining Context Free Grammars

A quadruple (V, Σ, R, S) , where

- ▶ V is an alphabet.
- ▶ Σ (the set of **terminals**) is a subset of V
- ▶ R (the set of **rules**) is a finite subset of $(V - \Sigma) \times V^*$, and
- ▶ S (the **start symbol**) is an element of $V - \Sigma$

We can describe so much more

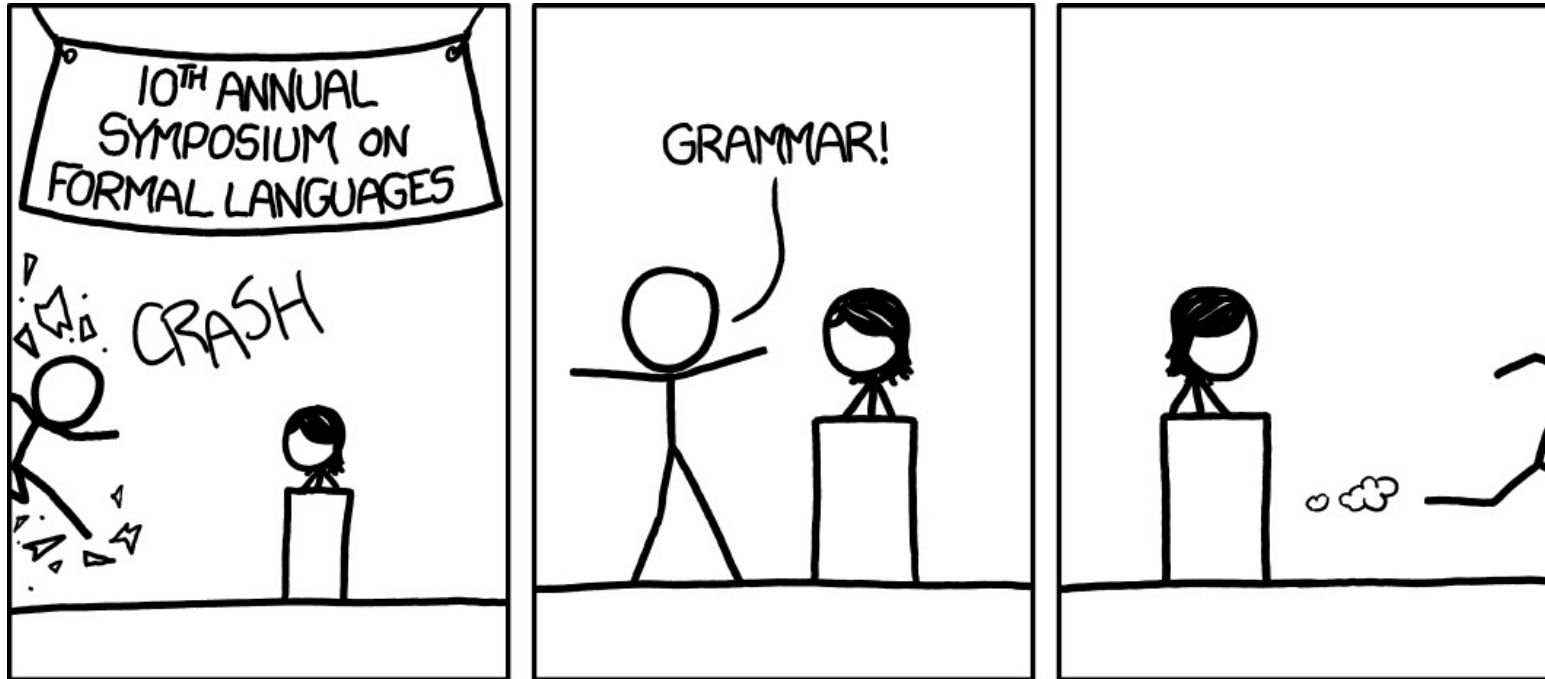
Give a CFG for $L = \{a^n b^n : n \geq 0\}$

Note: L is not a regular language!

$$\left. \begin{array}{l} S \rightarrow \varepsilon \\ S \rightarrow a S b \end{array} \right\} S \rightarrow \varepsilon \mid a S b$$

$$\varepsilon \in L \quad \wedge \quad [\forall w \quad w \in L \rightarrow a w b \in L]$$

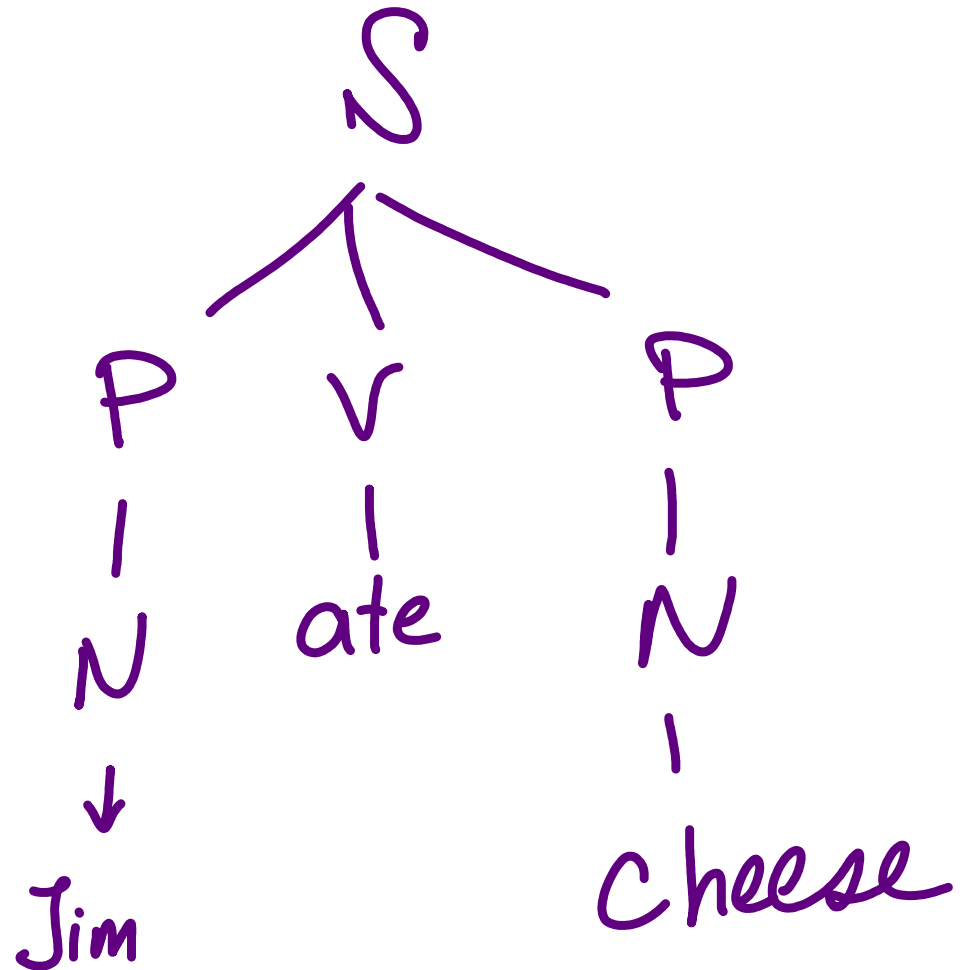
XKCD # 1090: Formal Languages



[audience looks around] 'What just happened?'
'There must be some context we're missing.'

English-looking strings

- ▶ $S \rightarrow PVP$
- ▶ $P \rightarrow N$
- ▶ $P \rightarrow AP$
- ▶ $A \rightarrow \text{big}$
- ▶ $A \rightarrow \text{green}$
- ▶ $N \rightarrow \text{cheese}$
- ▶ $N \rightarrow \text{Jim}$
- ▶ $V \rightarrow \text{ate}$



Computer Program Grammar

E, F, T stand for *expression, term, factor*.

▶ $E \rightarrow E + T$

▶ $E \rightarrow T$

▶ $T \rightarrow T * F$

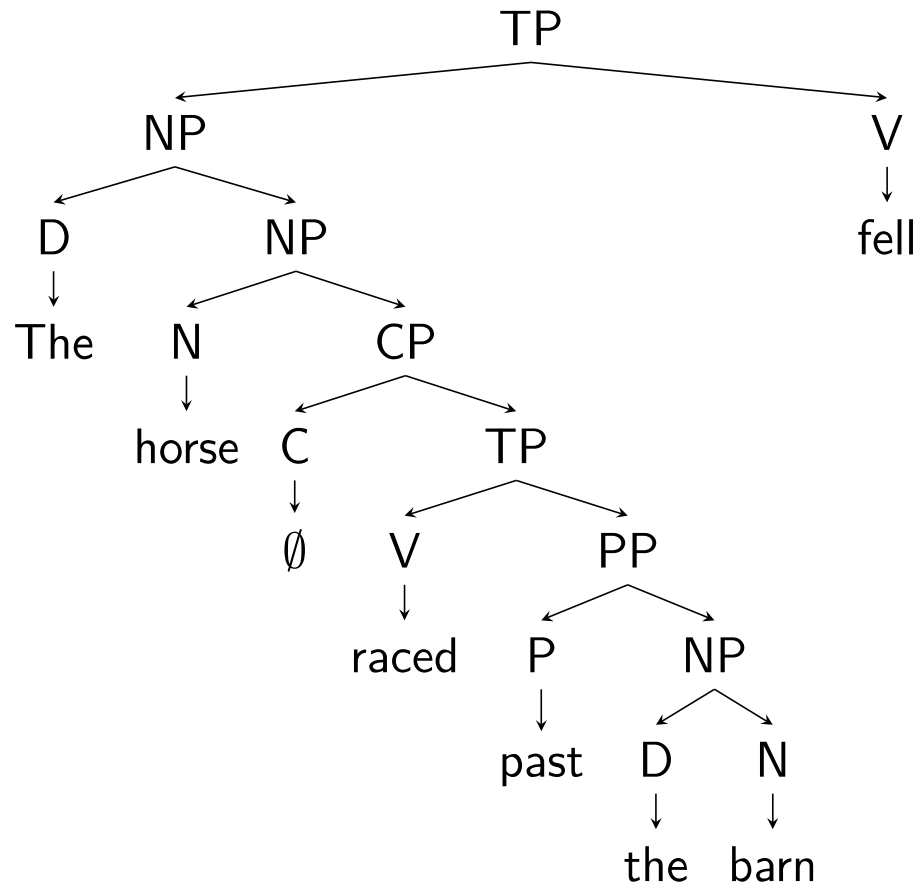
▶ $T \rightarrow F$

▶ $F \rightarrow (E)$

▶ $F \rightarrow \text{id}$

id * id + id
as parse tree?

Parse Trees



Balanced Parentheses

► $S \rightarrow \varepsilon \mid SS \mid (S)$

► Give a derivation of the string $()(())$

$$\begin{aligned}
 S &\rightarrow SS \rightarrow (S)S \rightarrow ()S \\
 &\rightarrow ()(S) \rightarrow ()(S) \\
 &\rightarrow ()(())
 \end{aligned}$$

$$\begin{aligned}
 S &\rightarrow SS \rightarrow S(S) \rightarrow S((S)) \rightarrow S(()) \\
 &\rightarrow (S)(()) \rightarrow ()(())
 \end{aligned}$$

Unambiguous Production

- ▶ Same language
- ▶ $S \rightarrow \underline{(RS} \mid \varepsilon$
- ▶ $R \rightarrow) \mid (RR$
- ▶ Production: look at next terminal.
 - ▶ If expand S , use $(RS$ if
 - ▶ If expand R , use $)$ if

Left to Right

Leftmost production

one token lookahead

LL(1)

() (())

$S \rightarrow \underline{(RS} \rightarrow () \underline{S} \rightarrow () (\underline{RS} \rightarrow () ((\underline{RRS}$