

Grammars

Language view:

A grammar is a set of structural rules that govern the composition of sentences, phrases and words.

Computational view:

A grammar (often called a "formal grammar") is a set of rules that describe what strings are valid in a formal language.

Grammars

What types of (formal) grammars have you heard of before?

Lots of different kinds of grammars:

- regular
- context-free
- context-sensitive
- recursively enumerable
- transformation grammars

Context Free Grammars (CFG)

How many people have heard of them?

What do you know about them?

Where are they used?

CFG production rules

$\mathsf{S} \to \mathsf{NP} \; \mathsf{VP}$

left hand side (single symbol)

right hand side (one or more symbols)

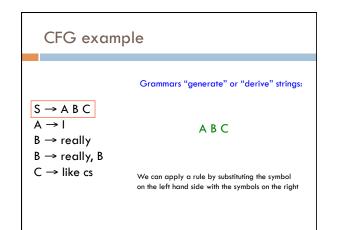
CFG example

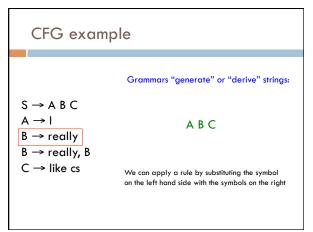
Grammars "generate" or "derive" strings:

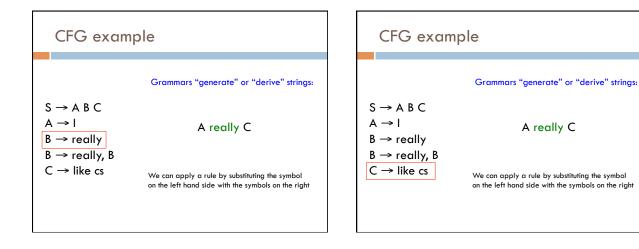
S

- $S \rightarrow A B C$
- $A \rightarrow I$
- $\textbf{B} \rightarrow \text{really}$
- $\textbf{B} \rightarrow \textbf{really, B}$
- $\mathsf{C} \twoheadrightarrow \mathsf{like}\ \mathsf{cs}$

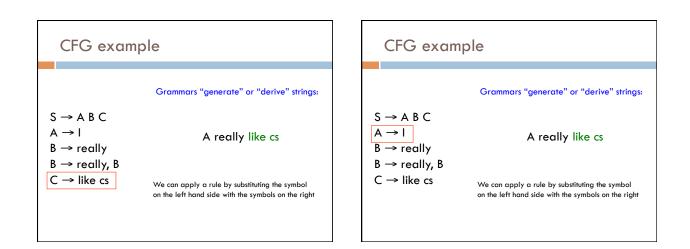
CFG example		
	Grammars "generate" or "derive" strings:	
$S \rightarrow A B C$ $A \rightarrow I$ $B \rightarrow really$	S	
$B \rightarrow really, B$ C \rightarrow like cs	We can apply a rule by substituting the symbol on the left hand side with the symbols on the right	

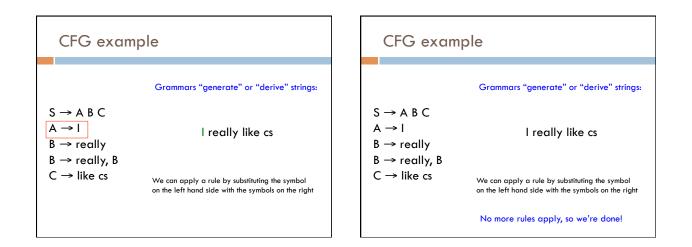


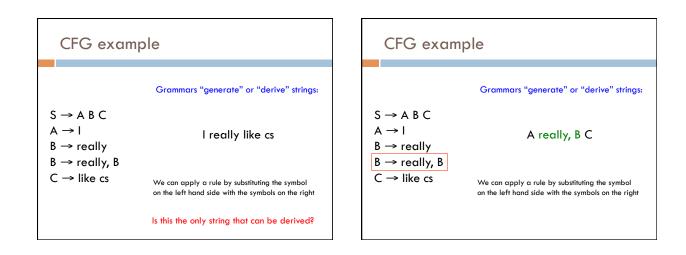




3







CFG example	CFG example
$\begin{array}{l} \mbox{Grammars "generate" or "derive" strings:} \\ S \rightarrow A \ B \ C \\ A \rightarrow I \\ B \rightarrow really \\ B \rightarrow really, B \\ C \rightarrow like \ cs \end{array} \qquad A \ really, really, B \ C \\ \ We \ can apply \ a \ rule \ by \ substituting \ the \ symbol son \ the \ right \end{array}$	$\begin{array}{c} Grammars \ describe \ a \ language, i.e. \ the strings (aka \ sentences) \ that \ are \ part \ of \ that \ language \ A \rightarrow I \\ B \rightarrow really \\ B \rightarrow really, B \qquad I \ really, really, really, \ like \ cs \\ C \rightarrow like \ cs \end{array}$

CFGs formally G = (NT, T, P, S)NT: finite set of nonterminal symbols T: finite set of terminal symbols, NT and T are disjoint P: finite set of productions of the form $A \rightarrow \alpha$, $A \in NT$ and $\alpha \in (T \cup NT)^*$ $S \in NT$: start symbol

What language does this represent? $S \rightarrow \alpha S$ $S \rightarrow E$ $E \rightarrow b E$ $E \rightarrow b$

