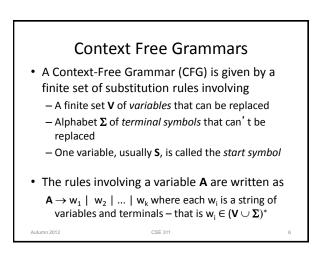
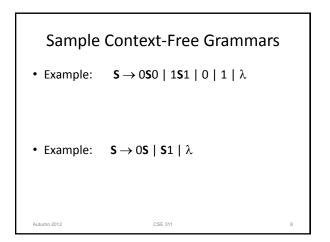


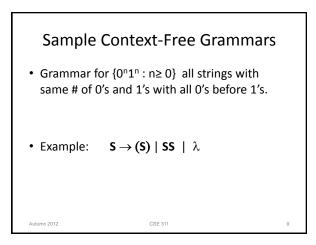
Regular expressions can't specify everything we might want • Even some easy things like palindromes • More complicated structures in programming languages – Matched parentheses – Properly formed arithmetic expressions – Etc.

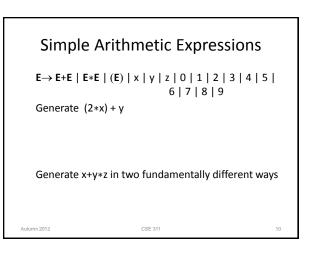
CSE 311



How Context-Free Grammars generate strings Begin with start symbol S • If there is some variable **A** in the current string you can replace it by one of the w's in the rules for A -Write this as $xAy \Rightarrow xwy$ - Repeat until no variables left • The set of strings the CFG generates are all strings produced in this way that have no variables CSE 311







Context-Free Grammars and recursively-defined sets of strings

- A CFG with the start symbol **S** as its only variable recursively defines the set of strings of terminals that **S** can generate
- A CFG with more than one variable is a simultaneous recursive definition of the sets of strings generated by *each* of its variables
 - Sometimes necessary to use more than one

CSE 311

11

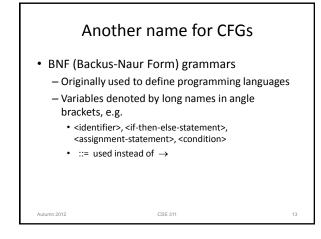
Building in Precedence in Simple Arithmetic Expressions

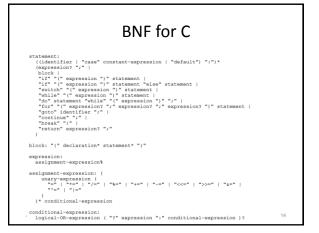
- E expression (start symbol)
- T term F factor I identifier N number $E{\rightarrow} T \mid E{+}T$ $T{\rightarrow} F \mid F{*}T$ $\mathbf{F} \rightarrow (\mathbf{E}) \mid \mathbf{I} \mid \mathbf{N}$
 - $I \rightarrow x \mid y \mid z$

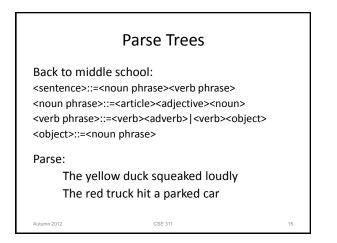
Autumn 2012

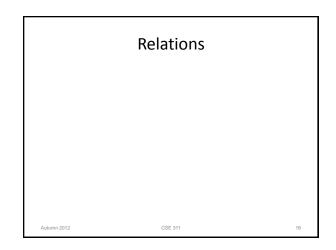
 $N \rightarrow 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

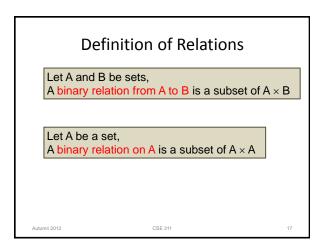
CSE 311

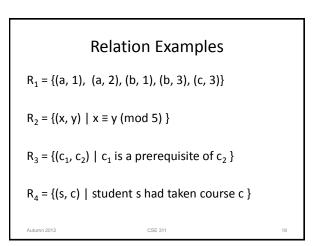












Properties of Relations

Let R be a relation on A

R is reflexive iff (a,a) \in R for every a \in A

R is symmetric iff $(a,b) \in R$ implies $(b, a) \in R$

R is antisymmetric iff (a,b) \in R and a \neq b implies (b,a) \notin R

R is transitive iff $(a,b) \in R$ and $(b, c) \in R$ implies $(a, c) \in R$

Autumn 2012

CSE 311

19