Grammars (Section 3.3)

Grammars

- A grammar is a finite set of rules, called *productions*, that are used to describe the strings of a language.
- *Notational Example:* The productions take the form $\alpha \to \beta$ where α and β are strings over an alphabet of *terminals* and *nonterminals*. Read $\alpha \to \beta$ as "α produces β" "α derives β" or "α is replaced by β". The following four expressions are productions for a grammar:
  - $S \to aSB$
  - $S \to \Lambda$
  - $B \to bB$
  - $B \to b$

Grammar terminology

First, an alternate short form for the previous grammar is:

- $S \to aSB | \Lambda$
- $B \to bB | b$.

Terminology:

- Terminals: $\{a, b\}$, the alphabet of the language.
- Nonterminals: $\{S, B\}$, the grammar symbols (uppercase letters), disjoint from terminals.
- Start symbol: S, a specified nonterminal alone on the left side of some production.
- Sentential form: any string of terminals and/or nonterminals.

Derivations

- Derivation: a transformation of sentential forms by means of productions as follows: if $x\alpha y$ is a sentential form and $\alpha \to \beta$ is a production, then the replacement of α by β in $x\alpha y$ to obtain $x\beta y$ is a *derivation step*, which we denote by $x\alpha y \Rightarrow x\beta y$.
- *Example Derivation:*
  $S \Rightarrow aSB \Rightarrow aaSBB \Rightarrow aaBB \Rightarrow aabBB \Rightarrow aabbB \Rightarrow aabbb$.
- This is a *leftmost derivation*, where each step replaces the leftmost nonterminal. The symbol $\Rightarrow^+$ means one or more steps and $\Rightarrow^*$ means zero or more steps. So we could write $S \Rightarrow^+ aabbb$ or $S \Rightarrow^* aabbb$ or $aSB \Rightarrow^* aSB$, and so on.

The Language of a Grammar

- The language of a grammar is the set of terminal strings derived from the start symbol.
- *Example:* Can we find the language of the grammar: $S \to aSB|\Lambda$ and $B \to bB|b$?

The Language of a Grammar

- The language of a grammar is the set of terminal strings derived from the start symbol.
- *Example:* Can we find the language of the grammar: $S \rightarrow aSB | \Lambda$ and $B \rightarrow bB | b$?
- *Solution:* Examine some derivations to see if a pattern emerges
  - $S \Rightarrow \Lambda$
  - $S \Rightarrow aSB \Rightarrow aB \Rightarrow ab$
  - $S \Rightarrow aSB \Rightarrow aB \Rightarrow abB \Rightarrow abbB \Rightarrow abbb$
  - $S \Rightarrow aSB \Rightarrow aaSBB \Rightarrow aaBB \Rightarrow aabB \Rightarrow aabb$
  - $S \Rightarrow aSB \Rightarrow aaSBB \Rightarrow aaBB \Rightarrow aabBB \Rightarrow aabbBB \Rightarrow aabbbB \Rightarrow aabbbb$

Solution

- So, we have a pretty good idea that the language of the grammar is:

Solution

- So, we have a pretty good idea that the language of the grammar is: $\{a^n b^{n+k} | n, k \in \mathbb{N}\}$
- *Quiz:* Describe the language of the grammar $S \rightarrow a | bcS$

Solution

- So, we have a pretty good idea that the language of the grammar is: $\{a^n b^{n+k} | n, k \in \mathbb{N}\}$
- *Quiz:* Describe the language of the grammar $S \rightarrow a | bcS$
- Solution: $\{(bc)^n a | n \in \mathbb{N}\}$.

Construction of Grammars

- *Example:* Find a grammar for $\{a^n b | n \in \mathbb{N}\}$

Construction of Grammars

- *Example:* Find a grammar for $\{a^n b | n \in \mathbb{N}\}$
- *Solution:* We need to derive any string of a's followed by b. The production $S \to aS$ can be used to derive strings of a's. The production $S \to b$ will stop the derivation and produce the desired string ending with b. So a grammar for the language is $S \to aS | b$.
- *Quiz:* Find a grammar for $\{ba^n | n \in \mathbb{N}\}$.

Construction of Grammars

- *Example:* Find a grammar for $\{a^n b | n \in \mathbb{N}\}$
- *Solution:* We need to derive any string of a's followed by b. The production $S \to aS$ can be used to derive strings of a's. The production $S \to b$ will stop the derivation and produce the desired string ending with b. So a grammar for the language is $S \to aS|b$.
- *Quiz:* Find a grammar for $\{ba^n | n \in \mathbb{N}\}$.
- *Solution:* $S \to Sa|b$.
- *Quiz:* Find a grammar for $\{(ab)^n | n \in \mathbb{N}\}$.

Construction of Grammars

- *Example:* Find a grammar for $\{a^n b | n \in \mathbb{N}\}$
- *Solution:* We need to derive any string of a's followed by b. The production $S \to aS$ can be used to derive strings of a's. The production $S \to b$ will stop the derivation and produce the desired string ending with b. So a grammar for the language is $S \to aS | b$.
- *Quiz:* Find a grammar for $\{ba^n | n \in \mathbb{N}\}$.
- *Solution:* $S \to Sa | b$.
- *Quiz:* Find a grammar for $\{(ab)^n | n \in \mathbb{N}\}$.
- *Solution:* $S \to Sab | \Lambda$ or $S \to abS | \Lambda$.

Rules for Combining Grammars

Let L and M be two languages with grammars that have start symbols A and B respectively, and with disjoint sets of nonterminals. Then the following rules apply:

- $L \cup M$ has a grammar starting with $S \to A|B$.
- LM has a grammar starting with $S \to AB$.
- $L*$ has a grammar starting with $S \to AS|\Lambda$.

Example: Find a grammar for $\{a^m b^m c^n | m, n \in \mathbb{N}\}$

Rules for Combining Grammars

Let L and M be two languages with grammars that have start symbols A and B respectively, and with disjoint sets of nonterminals. Then the following rules apply:

- $L \cup M$ has a grammar starting with $S \to A|B$.
- LM has a grammar starting with $S \to AB$.
- $L*$ has a grammar starting with $S \to AS|\Lambda$.

Example: Find a grammar for $\{a^m b^m c^n | m, n \in \mathbb{N}\}$ *Solution:* The language is the product LM, where $L = \{a^m b^m | m \in \mathbb{N}\}$ and $M = \{c^n | n \in \mathbb{N}\}$. So a grammar for LM can be written in terms of grammars for L and M as follows:

- $S \to AB$
- $A \to aAb|\Lambda$
- $B \to cB|\Lambda$.

Inductive definitions

Example: Find a grammar for the language L defined inductively by:

- Basis: $a, b, c \in L$
- Induction: If $x, y \in L$ then $f(x), g(x, y) \in L$

Solution: We can get some idea about L by listing some of its strings.

- $a, b, c, f(a), f(b), \ldots, g(a, a), \ldots, g(f(a), f(a)), \ldots, f(g(b, c)), \ldots, g$

So L is the set of all algebraic expressions made up from the letters a, b, c and the function symbols f and g of arities 1 and 2, respectively. A grammar for L can be written as:

- $S \rightarrow a|b|c|f(S)|g(S, S)$.

Example derivation

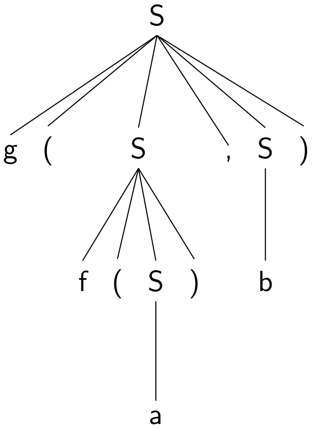
For example, a leftmost derivation of $g(f(a), g(b, f(c)))$ can be written as:

- $S \Rightarrow g(S, S) \Rightarrow g(f(S), S) \Rightarrow g(f(a), S) \Rightarrow g(f(a), g(S, S)) \Rightarrow g(f(a), g(b, S)) \Rightarrow g(f(a), g(b, f(S))) \Rightarrow g(f(a), g(b, f(c)))$.

Parse Trees

- A Parse Tree is a tree that represents a derivation. The root is the start symbol and the children of a nonterminal node are the symbols (terminals, nonterminals, or Λ) on the right side of the production used in the derivation step that replaces that node.
- *Example:* The tree shown in the next slide is the parse tree for the following derivation:
  - $S \Rightarrow g(S, S) \Rightarrow g(f(S), S) \Rightarrow g(f(a), S) \Rightarrow g(f(a), b)$.

Parse Tree



Ambiguous Grammar

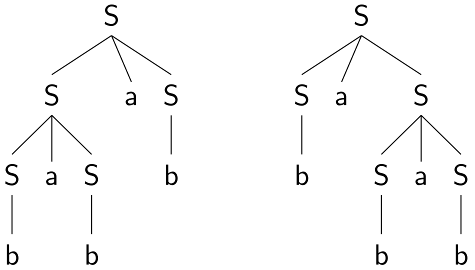
- The term ambiguous grammar means that there is at least one string with two distinct parse trees, or equivalently, two distinct leftmost derivations or two distinct rightmost derivations.
- *Example:* Is the grammar $S \rightarrow SaS|b$ ambiguous?

Ambiguous Grammar

- The term ambiguous grammar means that there is at least one string with two distinct parse trees, or equivalently, two distinct leftmost derivations or two distinct rightmost derivations.
- *Example:* Is the grammar $S \rightarrow SaS|b$ ambiguous?
- *Solution:* Yes. For example, the string *babab* has two distinct leftmost derivations:
    - $S \Rightarrow SaS \Rightarrow SaSaS \Rightarrow baSaS \Rightarrow babaS \Rightarrow babab$
    - $S \Rightarrow SaS \Rightarrow baS \Rightarrow baSaS \Rightarrow babaS \Rightarrow babab$

Ambiguous Parse Trees

- This ambiguity is perhaps best shown through the distinct parse trees:

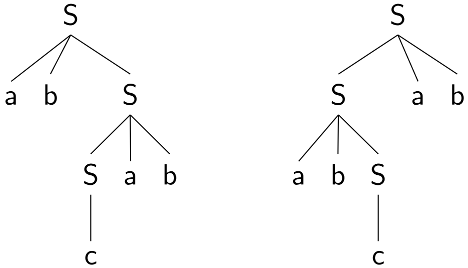


Another example

- Show that the grammar $S \to abS|Sab|c$ is ambiguous:

Another example

- Show that the grammar $S \to abS|Sab|c$ is ambiguous:



Unambiguous Grammars

- Sometimes you can find a grammar that is not ambiguous for the language of an ambiguous grammar.
- *Example:* The previous example showed $S \to SaS|b$ is ambiguous. The languages of the grammar is $\{b, bab, babab, \ldots\}$. Another grammar for the language is $S \to baS|b$. It is unambiguous because S produces either baS or b, which can't derive the same string.
- *Example:* Previously we showed $S \to c|abS|Sab$ is ambiguous. Its language is $\{(ab)^m c(ab)^n | m, n \in \mathbb{N}\}$. Another grammar for the language is: $S \to abS|cT$ and $T \to abT|\Lambda$.