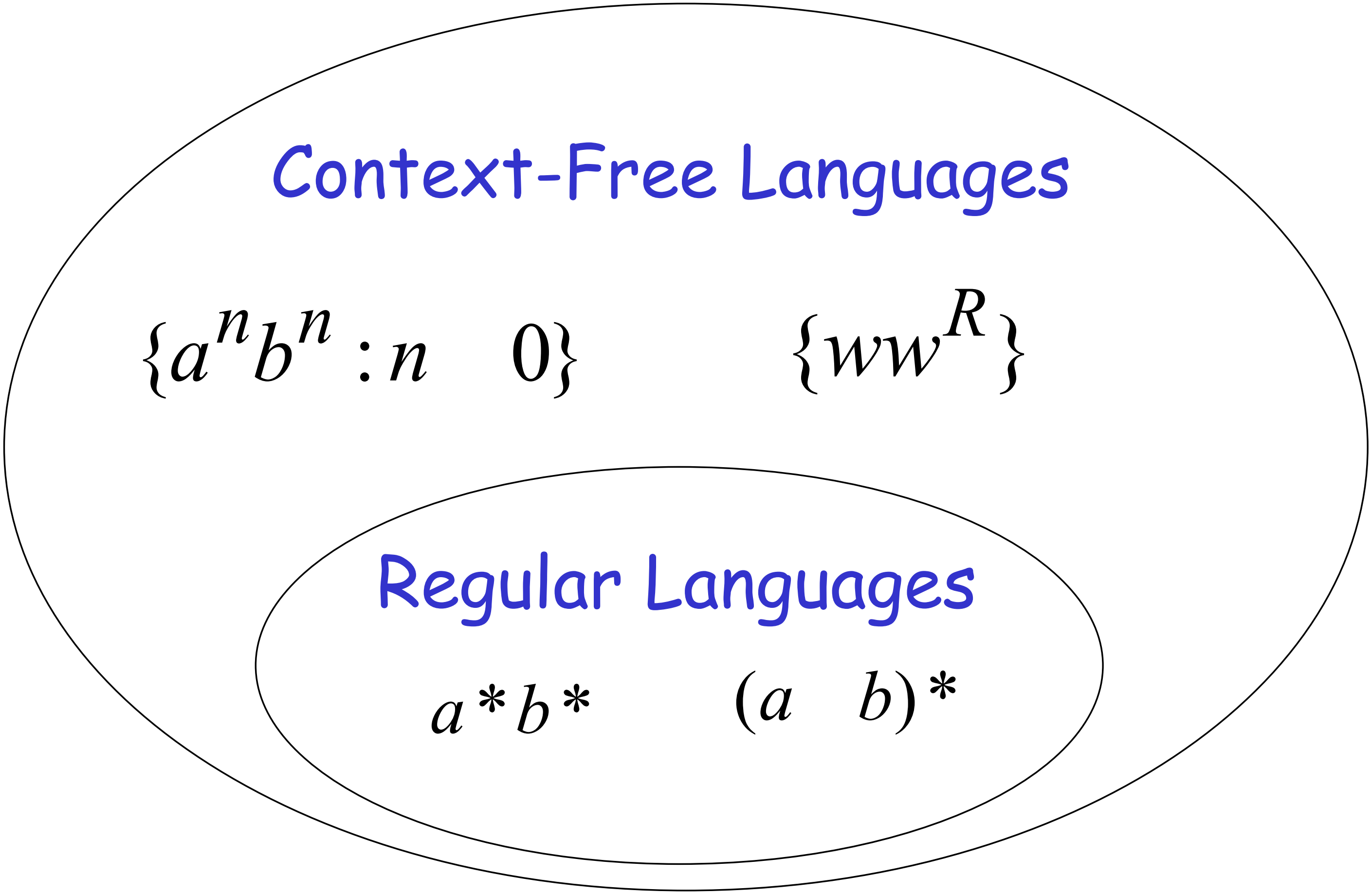
Context-Free Languages



Context-Free Languages

$\{a^n b^n : n \quad 0\}$

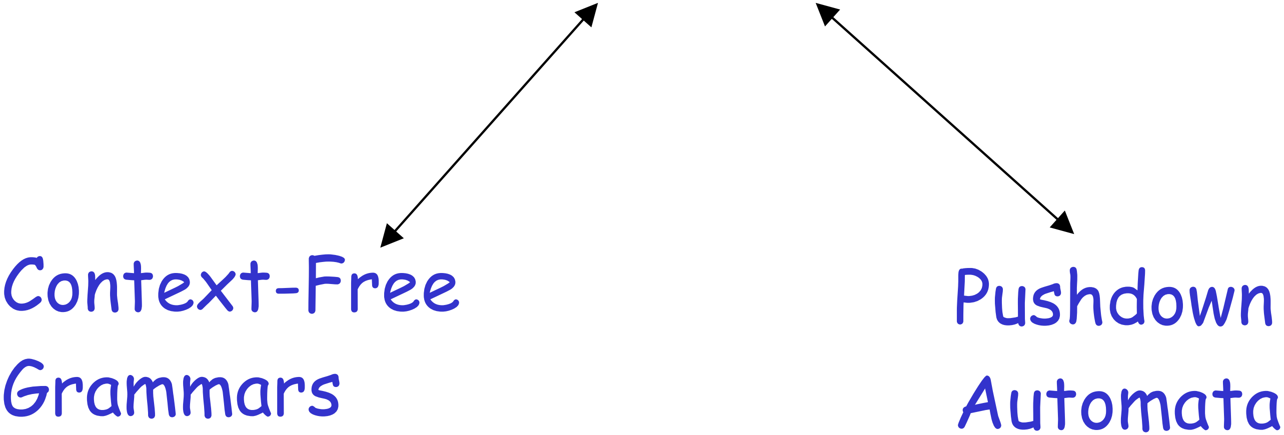
$\{ww^R\}$

Regular Languages

a^*b^*

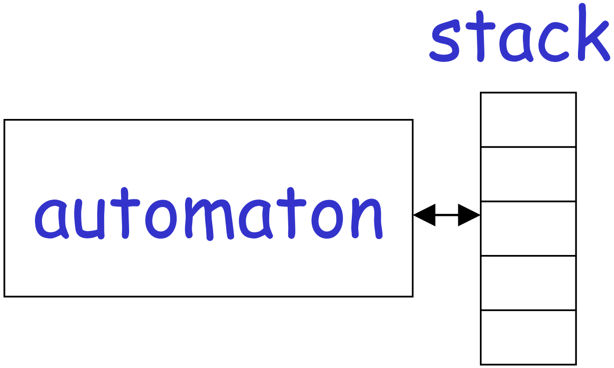
$(a \quad b)^*$

Context-Free Languages



Context-Free Grammars

Pushdown Automata



Context-Free Grammars

Grammars

Grammars express languages

Example: the English language grammar

$$\langle sentence \rangle \qquad \langle noun\_phrase \rangle \langle predicate \rangle$$

$$\langle noun\_phrase \rangle \qquad \langle article \rangle \langle noun \rangle$$

$$\langle predicate \rangle \qquad \langle verb \rangle$$

$\langle \mathit{article} \rangle$	*a*
$\langle \mathit{article} \rangle$	*the*
$\langle \mathit{noun} \rangle$	*cat*
$\langle \mathit{noun} \rangle$	*dog*
$\langle \mathit{verb} \rangle$	*runs*
$\langle \mathit{verb} \rangle$	*sleeps*

Derivation of string “the dog walks”:

$$
\begin{aligned}
\langle sentence \rangle \quad & \langle noun\_phrase \rangle \ \langle predicate \rangle \\
& \langle noun\_phrase \rangle \ \langle verb \rangle \\
& \langle article \rangle \ \langle noun \rangle \ \langle verb \rangle \\
& the \ \langle noun \rangle \ \langle verb \rangle \\
& the \ dog \ \langle verb \rangle \\
& the \ dog \ sleeps
\end{aligned}
$$

Derivation of string “a cat runs”:

$$\langle sentence \rangle \qquad \langle noun\_phrase \rangle \ \langle predicate \rangle$$

$$\langle noun\_phrase \rangle \ \langle verb \rangle$$

$$\langle article \rangle \ \langle noun \rangle \ \langle verb \rangle$$

$$a \ \langle noun \rangle \ \langle verb \rangle$$

$$a \ \ cat \ \langle verb \rangle$$

$$a \ \ cat \ \ runs$$

Language of the grammar:

L = { “a cat runs“,
“a cat sleeps“,
“the cat runs“,
“the cat sleeps“,
“a dog runs“,
“a dog sleeps“,
“the dog runs“,
“the dog sleeps“ }

Productions

$$\langle noun \rangle \qquad \overbrace{cat}^{\text{Sequence of Terminals (symbols)}}$$

$$\langle sentence \rangle \qquad \underbrace{\langle noun\_phrase \rangle \langle predicate \rangle}_{\text{Sequence of Variables}}$$

Variables

Another Example

Grammar:

$$S \qquad \overbrace{aSb}^{\text{Sequence of terminals and variables}}$$

$$S$$

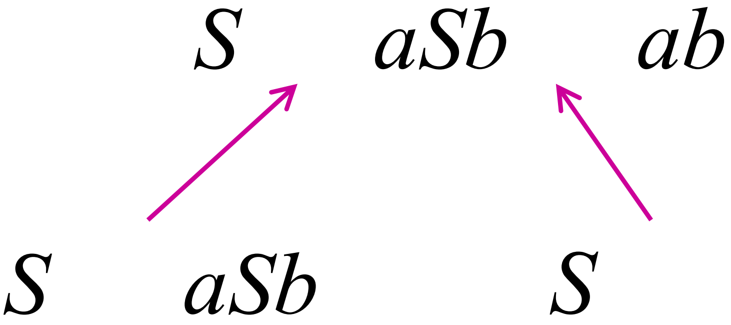
Variable

The right side may be

Grammar: $S \quad aSb$

S

Derivation of string ab:



$S \quad aSb \quad ab$

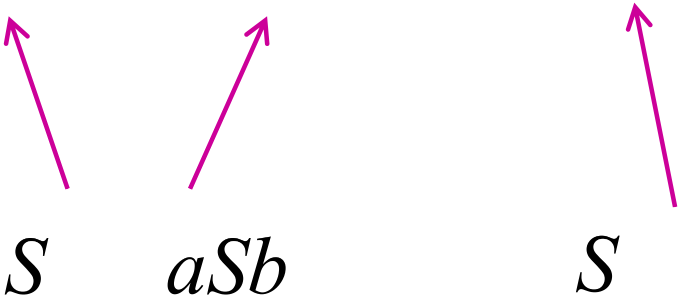
$S \quad aSb \qquad\qquad S$

Grammar: $S \quad aSb$

S

Derivation of string $aabb$:

$$S \quad aSb \quad aaSbb \quad aabb$$



$S \quad aSb$ $\qquad$ S

Grammar: $S \quad aSb$

S

Other derivations:

$$S \quad aSb \quad aaSbb \quad aaaSbbb \quad aaabbb$$

$$S \quad aSb \quad aaSbb \quad aaaSbbb$$
$$aaaaSbbbb \quad aaaabbbb$$

Grammar: $S \quad aSb$

S

Language of the grammar:

$$L \quad \{a^n b^n : n \quad 0\}$$

A Convenient Notation

We write: $S \overset{*}{\qquad} aaabbb$

for zero or more derivation steps

Instead of:

$$S \qquad aSb \qquad aaSbb \qquad aaaSbbb \qquad aaabbb$$

In general we write: $w_1 \quad \overset{*}{\quad} \quad w_n$

If: $w_1 \quad w_2 \quad w_3 \quad \quad w_n$

in zero or more derivation steps

Trivially: $w \quad \overset{*}{\quad} \quad w$

Example Grammar

$S \quad aSb$

S

Possible Derivations

$*$

S

$*$

$S \quad ab$

$*$

$S \quad aaabbb$

$S \quad aaSbb \quad aaaaaSbbbbb \;\; b$

Another convenient notation:

$$S \quad aSb$$
$$S$$

⇒

$$S \quad aSb \;|$$

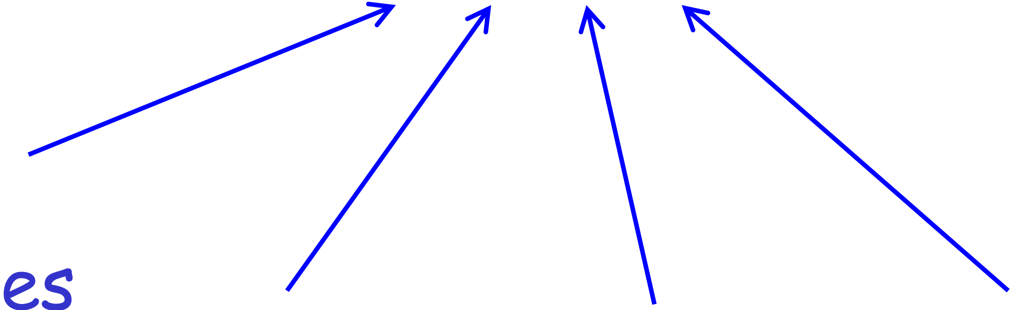
$$\langle article \rangle \quad a$$
$$\langle article \rangle \quad the$$

⇒

$$\langle article \rangle \quad a \,|\, the$$

Formal Definitions

Grammar: $G \quad V, T, S, P$



Set of variables

Set of terminal symbols

Start variable

Set of productions

Context-Free Grammar: $G \quad (V,T,S,P)$

All productions in P are of the form

$$A \qquad s$$

Variable

String of variables and terminals

Example of Context-Free Grammar

$$\boxed{S \qquad aSb \;\mid\; }$$

productions

$$P \quad \{S \qquad aSb, \;\; S \qquad \}$$

$$G \quad V, T, S, P$$

$$V \quad \{S\}$$

variables

$$T \quad \{a, b\}$$

terminals

start variable

Language of a Grammar:

For a grammar G with start variable S

$$L(G) \quad \{w : \quad S \quad \overset{*}{} \quad w, \quad w \quad T*\}$$

String of terminals or

Example:

context-free grammar G : $\boxed{S \qquad aSb \mid \quad}$

$$L(G) \qquad \{a^n b^n : \; n \qquad 0\}$$

Since, there is derivation

$S \qquad a^n b^n$ for any $n \qquad 0$

Context-Free Language:

A language L is context-free

if there is a context-free grammar G

with $L \quad L(G)$

Example:

$$L \quad \{a^n b^n : \quad n \quad 0\}$$

is a context-free language

since context-free grammar G :

$$\boxed{S \qquad aSb \mid}$$

generates $L(G) \quad L$

Another Example

Context-free grammar G :

$$S \qquad aSa \mid bSb \mid$$

Example derivations:

$$S \quad aSa \quad abSba \quad abba$$

$$S \quad aSa \quad abSba \quad abaSaba \quad abaaba$$

$$L(G) \quad \{ww^R : \; w \quad \{a,b\}^*\}$$

Palindromes of even length

Another Example

Context-free grammar G:

$$S \qquad aSb \mid SS \mid$$

Example derivations:

$$S \quad SS \quad aSbS \quad abS \quad ab$$

$$S \quad SS \quad aSbS \quad abS \quad abaSb \quad abab$$

$$L(G) \qquad \{w \quad : n_a(w) \quad n_b(w),$$

$$\text{and } n_a(v) \quad n_b(v)$$

$$\text{in any prefix } v\}$$

Describes matched parentheses: () ((( ))) (( )) a (,   b)

Derivation Order
and
Derivation Trees

Derivation Order

Consider the following example grammar with 5 productions:

1. S	AB	2. A	aaA	4. B	Bb
		3. A		5. B	

$$1.\ S \quad AB \qquad 2.\ A \quad aaA \qquad 4.\ B \quad Bb$$

$$3.\ A \qquad\qquad 5.\ B$$

Leftmost derivation order of string aab :

$$S \overset{1}{\quad} AB \overset{2}{\quad} aaAB \overset{3}{\quad} aaB \overset{4}{\quad} aaBb \overset{5}{\quad} aab$$

At each step, we substitute the leftmost variable

$$1.\ S \qquad AB \qquad 2.\ A \qquad aaA \qquad 4.\ B \qquad Bb$$

$$3.\ A \qquad\qquad 5.\ B$$

Rightmost derivation order of string aab :

$$S \overset{1}{\quad} AB \overset{4}{\quad} ABb \overset{5}{\quad} Ab \overset{2}{\quad} aaAb \overset{3}{\quad} aab$$

At each step, we substitute the rightmost variable

$$1.\ S \qquad AB \qquad 2.\ A \qquad aaA \qquad 4.\ B \qquad Bb$$
$$3.\ A \qquad\qquad 5.\ B$$

Leftmost derivation of aab :

$$S \overset{1}{\qquad} AB \overset{2}{\qquad} aaAB \overset{3}{\qquad} aaB \overset{4}{\qquad} aaBb \overset{5}{\qquad} aab$$

Rightmost derivation of aab :

$$S \overset{1}{\qquad} AB \overset{4}{\qquad} ABb \overset{5}{\qquad} Ab \overset{2}{\qquad} aaAb \overset{3}{\qquad} aab$$

Derivation Trees

Consider the same example grammar:

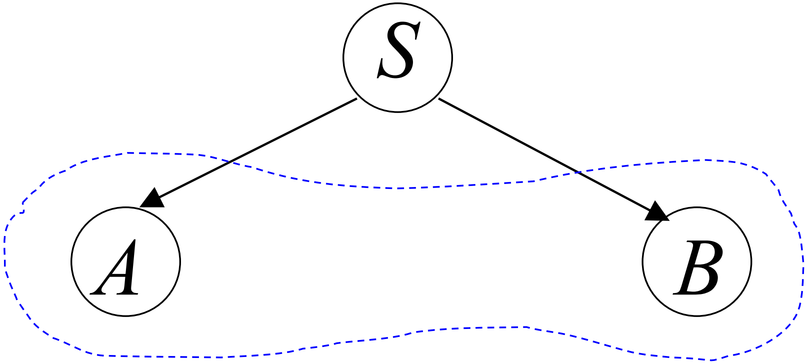
$$S \qquad AB \qquad\qquad A \qquad aaA \mid \qquad\qquad B \qquad Bb \mid$$

And a derivation of aab :

$$S \qquad AB \qquad aaAB \qquad aaABb \qquad aaBb \qquad aab$$

$S \quad AB \qquad A \quad aaA \mid \qquad B \quad Bb \mid$

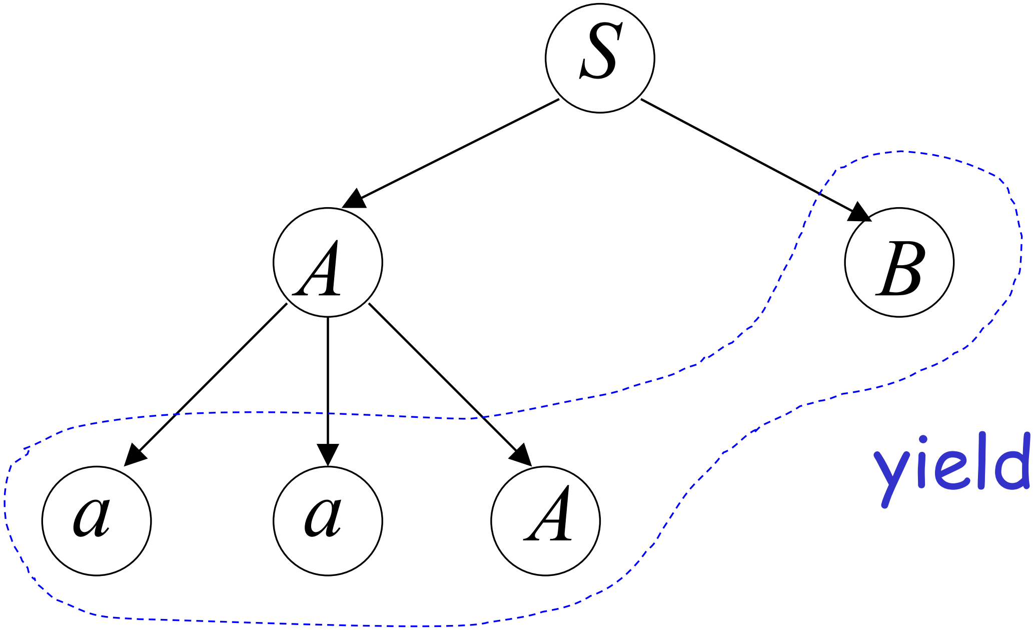
$S \quad AB$



yield AB

$S \quad AB \qquad A \quad aaA| \qquad B \quad Bb|$

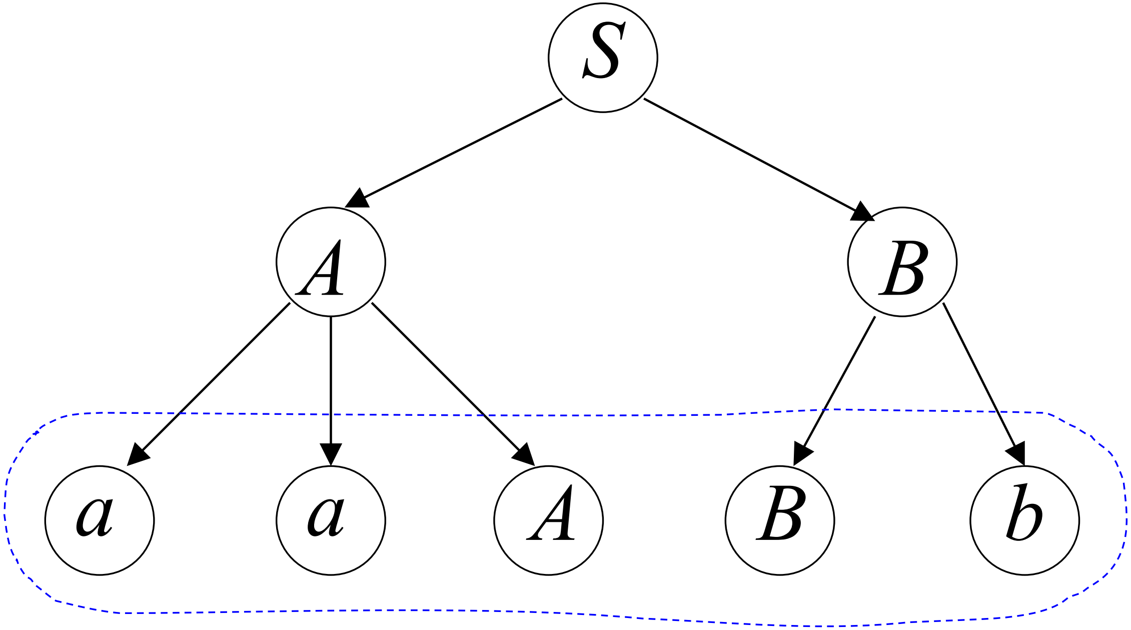
$S \quad AB \quad aaAB$



yield $aaAB$

$S \quad AB \qquad A \quad aaA \mid \qquad B \quad Bb \mid$

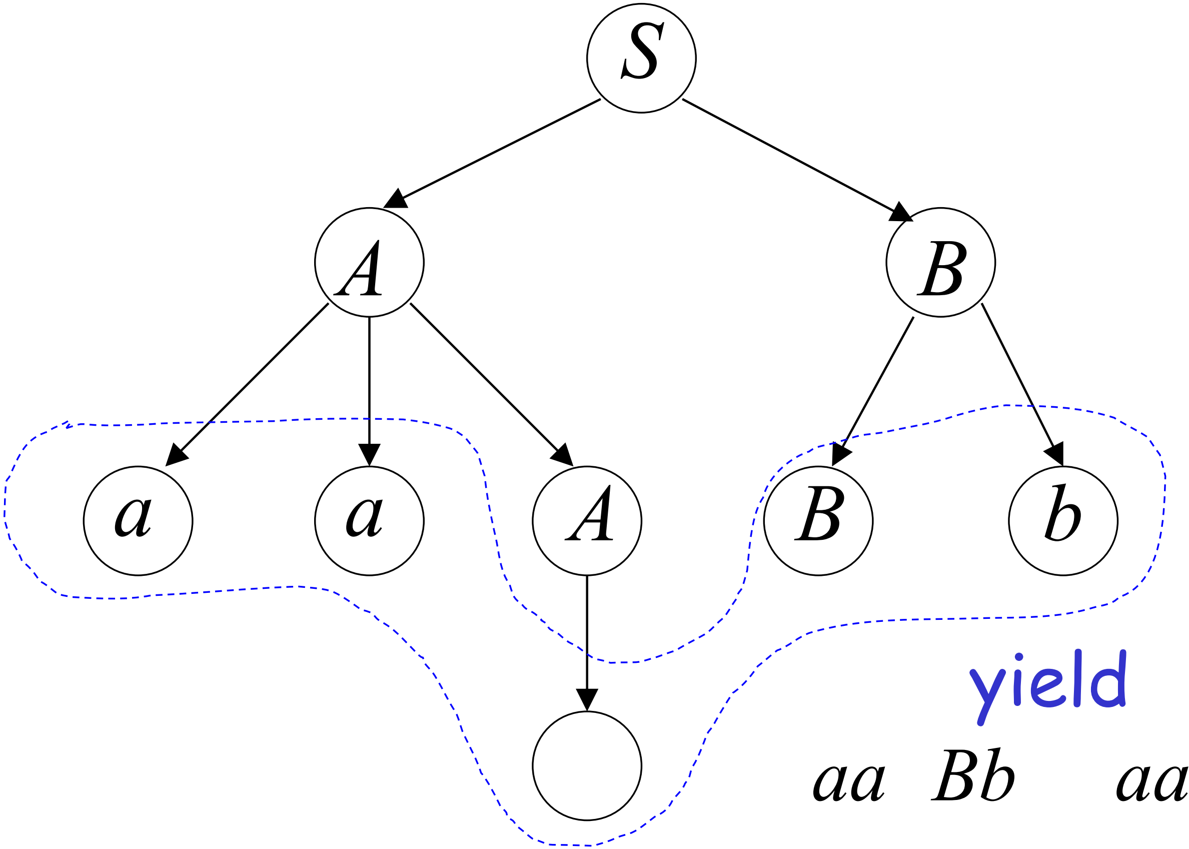
$S \quad AB \quad aaAB \quad aaABb$



yield $aaABb$

$S \quad AB \qquad A \quad aaA \mid \qquad B \quad Bb \mid$

$S \quad AB \quad aaAB \quad aaABb \quad aaBb$

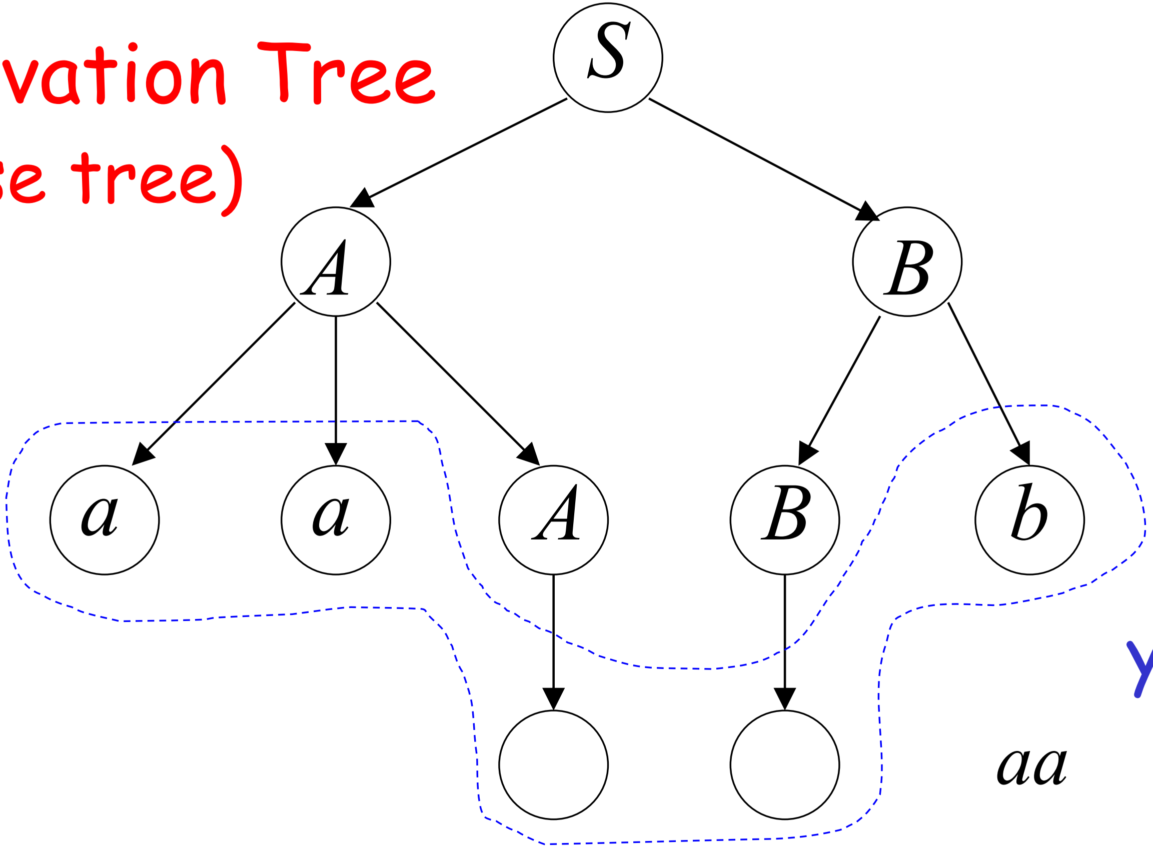


$aa \;\; Bb \qquad aaBb$

$S \qquad AB \qquad\qquad A \qquad aaA \mid \qquad\qquad B \qquad Bb \mid$

$S \qquad AB \qquad aaAB \qquad aaABb \qquad aaBb \qquad aab$

Derivation Tree (parse tree)



yield

$aa \quad b \quad aab$

Sometimes, derivation order doesn't matter

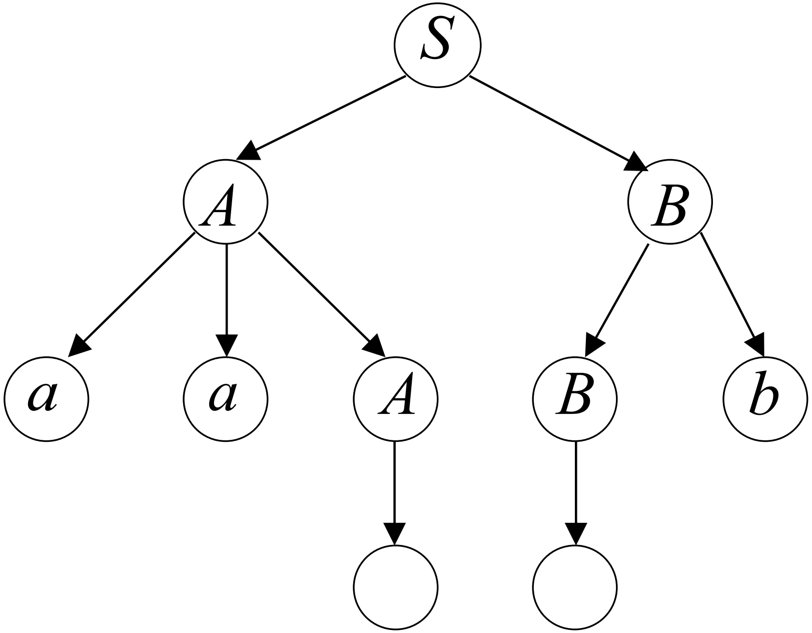
Leftmost derivation:

$S \qquad AB \qquad aaAB \qquad aaB \qquad aaBb \qquad aab$

Rightmost derivation:

$S \qquad AB \qquad ABb \qquad Ab \qquad aaAb \qquad aab$

Give same derivation tree



Ambiguity

Grammar for mathematical expressions

$$\boxed{E \qquad E \quad E \mid E \quad E \mid (E) \mid a}$$

Example strings:

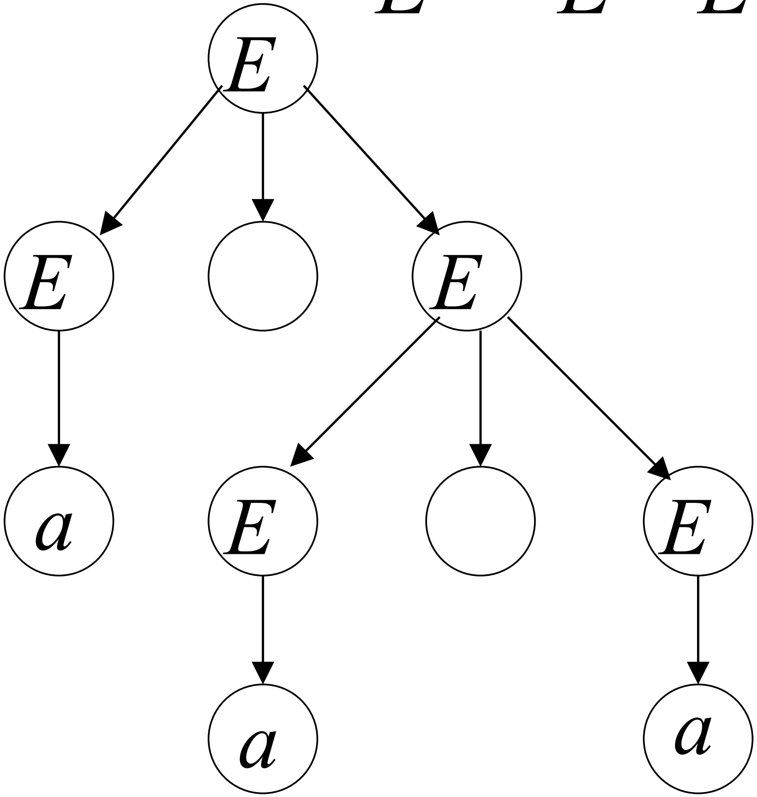
$$(a \quad a) \quad a \quad (a \quad a \quad (a \quad a))$$

Denotes any number

$$\boxed{E \qquad E \quad E \mid E \quad E \mid (E) \mid a}$$

$$E \qquad E \quad E \qquad a \quad E \qquad a \quad E \quad E$$

$$a \quad a \quad E \qquad a \quad a * a$$



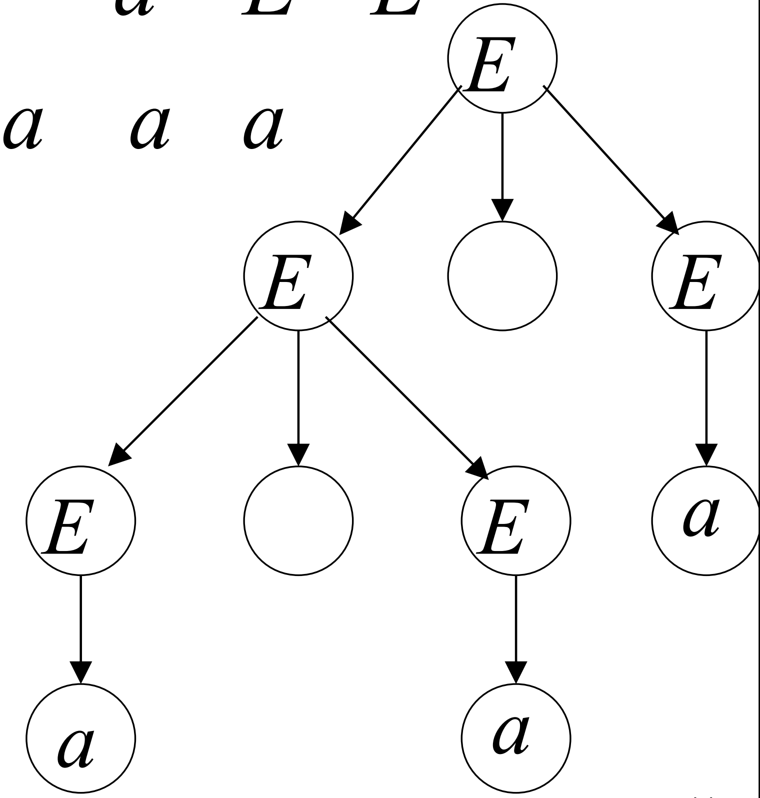
A leftmost derivation

for $a \quad a \quad a$

$$E \quad E \quad E \mid E \quad E \mid (E) \mid a$$

$$E \quad E \; E \quad E \; E \; E \quad a \; E \; E$$

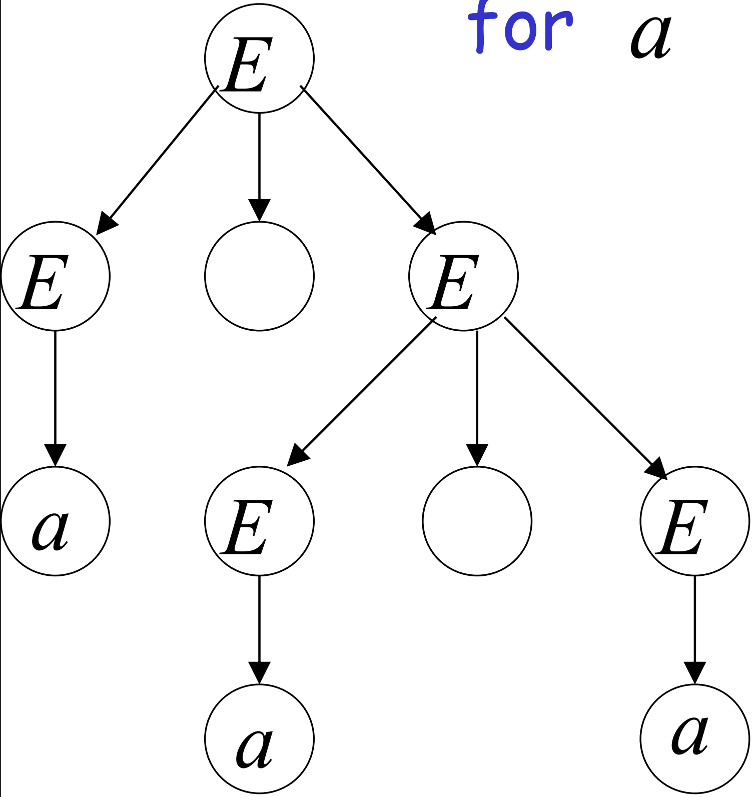
$$a \; a \; E \quad a \; a \; a$$

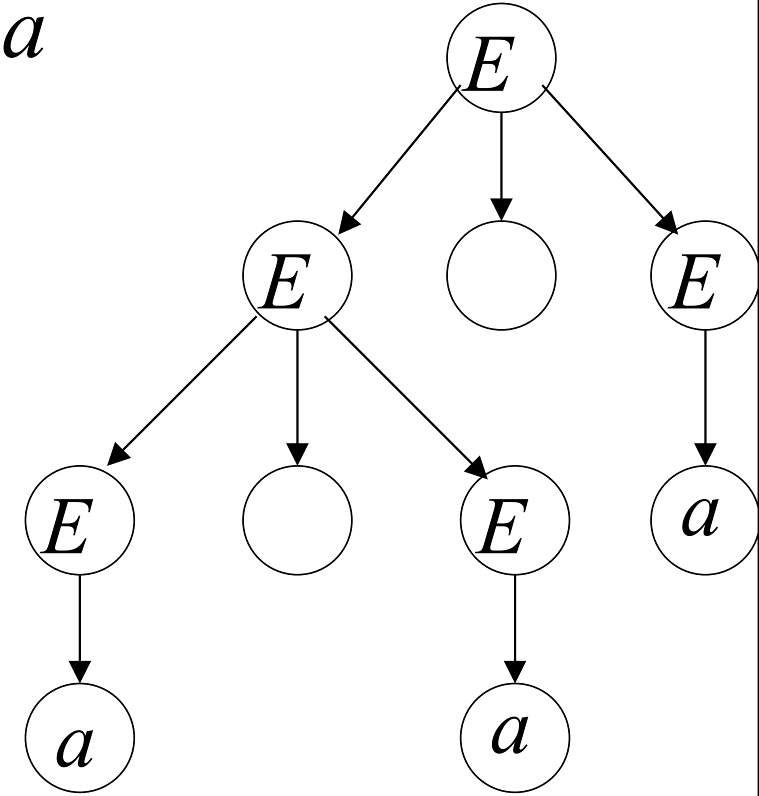


Another leftmost derivation for $a \quad a \quad a$

$E \quad E \quad E \mid E \quad E \mid (E) \mid a$

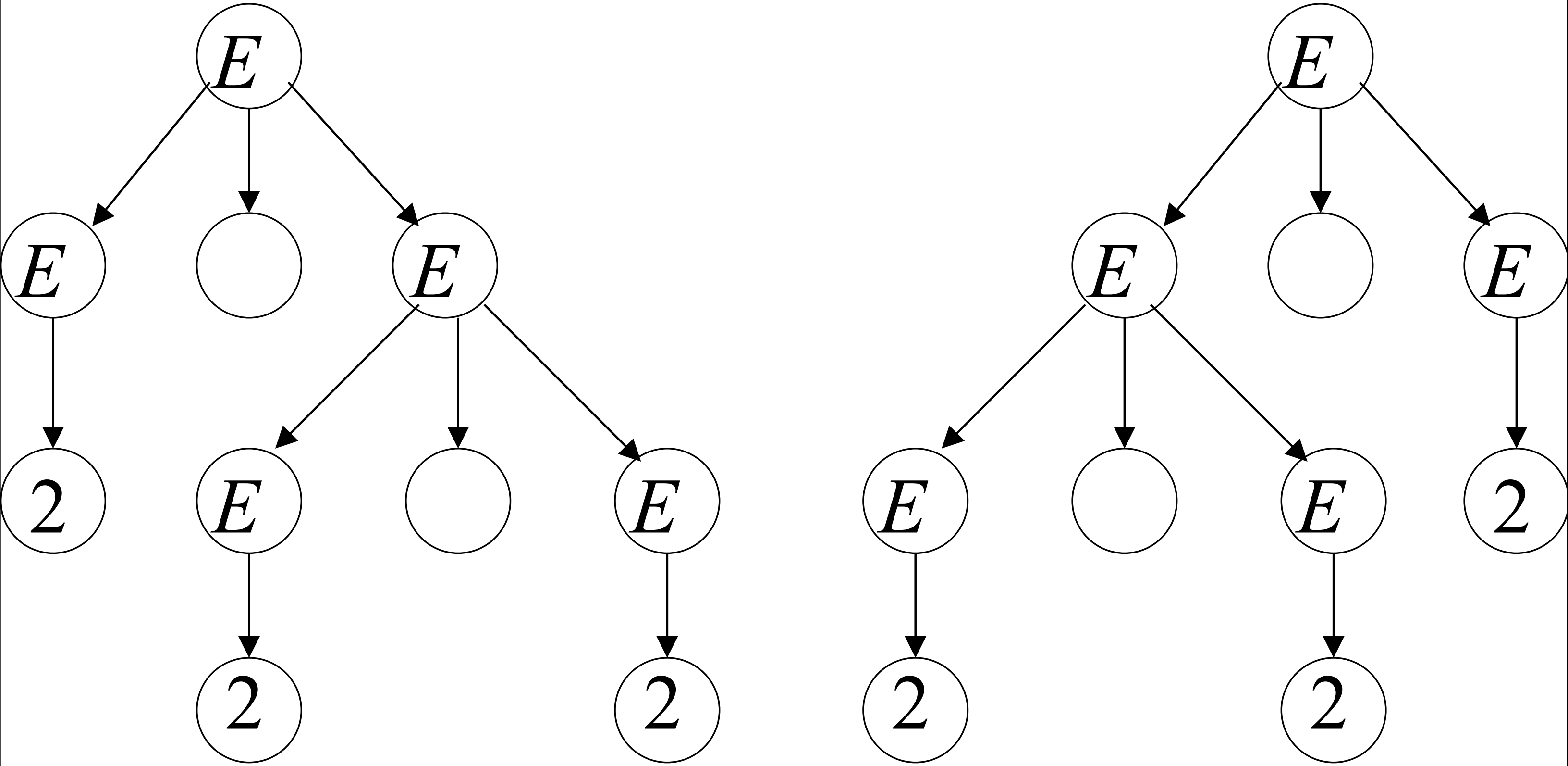
Two derivation trees for $a \quad a \quad a$





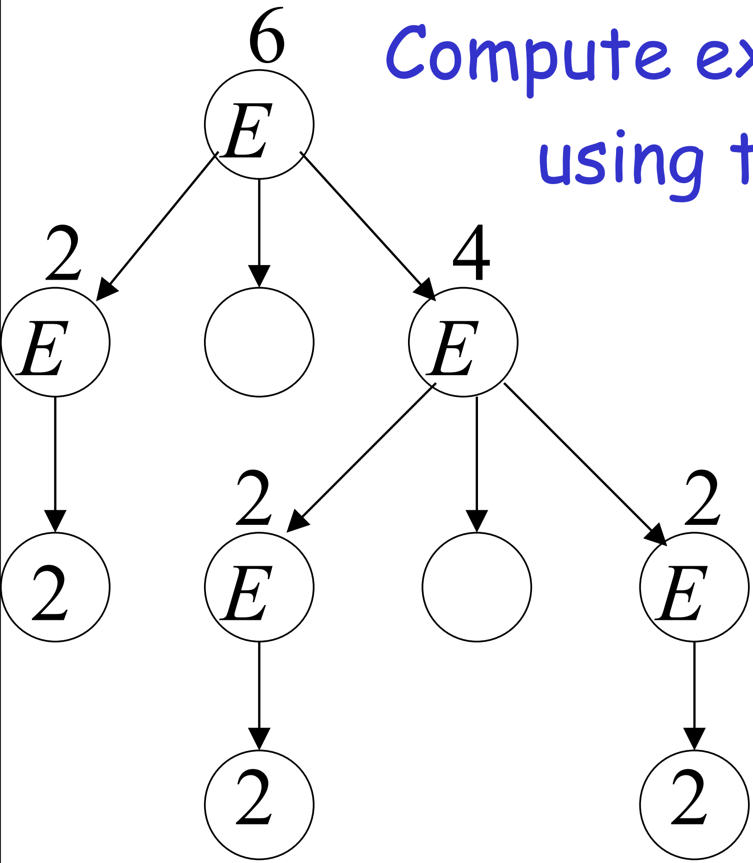
take $a \quad 2$

$a \quad a \quad a \quad 2 \quad 2 \quad 2$



Good Tree

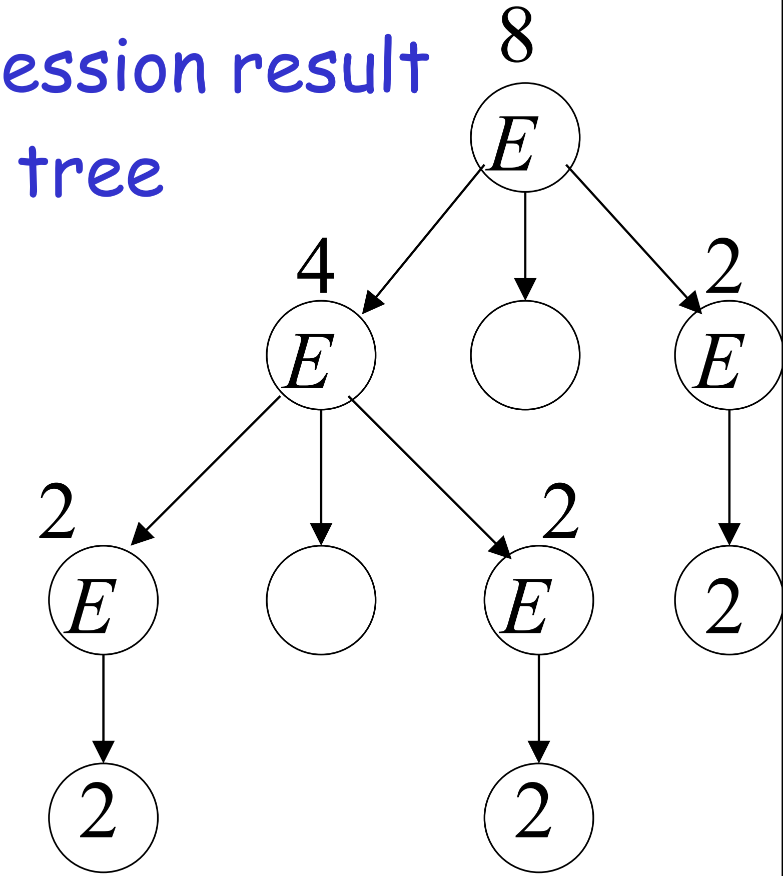
2 2 2 6



Compute expression result using the tree

Bad Tree

2 2 2 8



Two different derivation trees
may cause problems in applications which
use the derivation trees:

- Evaluating expressions

- In general, in compilers
  for programming languages

Ambiguous Grammar:

A context-free grammar G is ambiguous if there is a string w $L(G)$ which has:

two different derivation trees

or

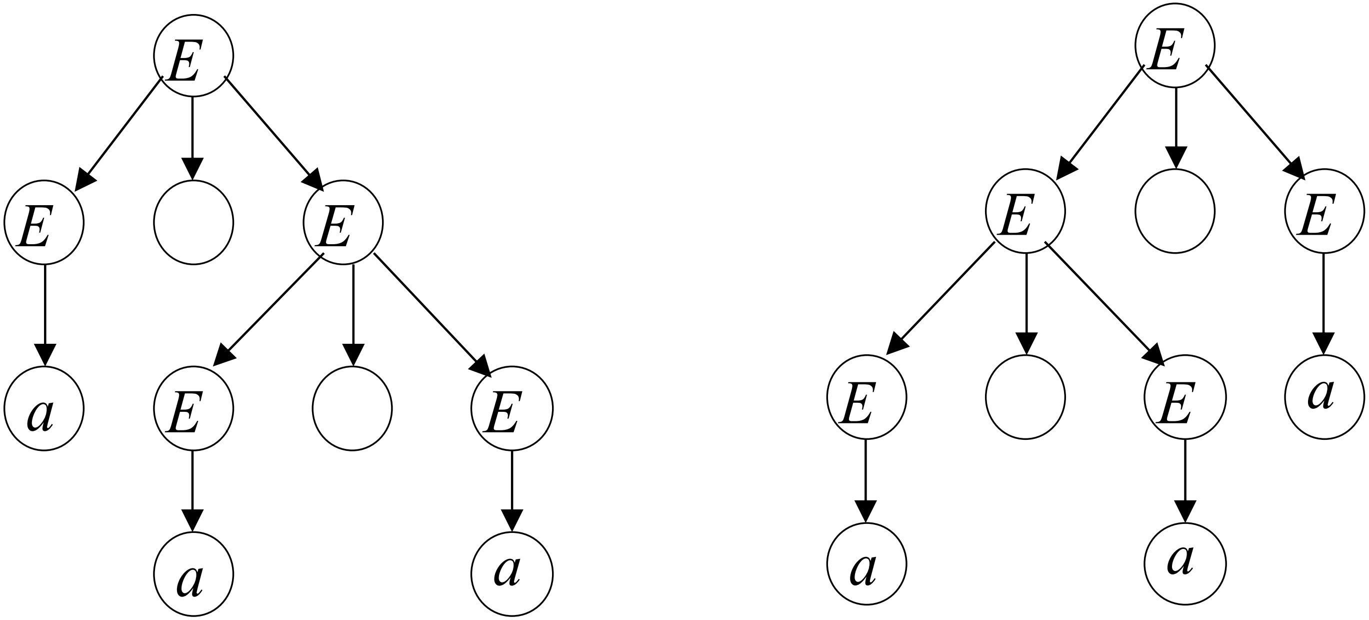
two leftmost derivations

(Two different derivation trees give two different leftmost derivations and vice-versa)

Example: $E \qquad E \quad E \mid E \quad E \mid (E) \mid a$

this grammar is ambiguous since

string $a \quad a \quad a$ has two derivation trees



$$\boxed{E \qquad E \quad E \mid E \quad E \mid (E) \mid a}$$

this grammar is ambiguous also because

string $a \quad a \quad a$ has two leftmost derivations

$$E \qquad E \quad E \qquad a \quad E \qquad a \quad E \quad E$$

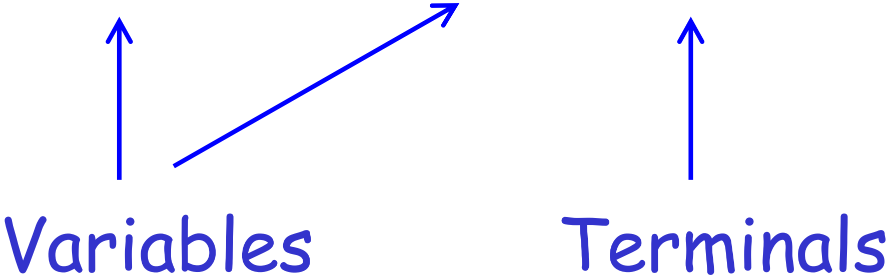
$$a \quad a \quad E \qquad a \quad a * a$$

$$E \qquad E \quad E \qquad E \quad E \quad E \qquad a \quad E \quad E$$

$$a \quad a \quad E \qquad a \quad a \quad a$$

Another ambiguous grammar:

IF_STMT   if EXPR then STMT
    | if EXPR then STMT else STMT

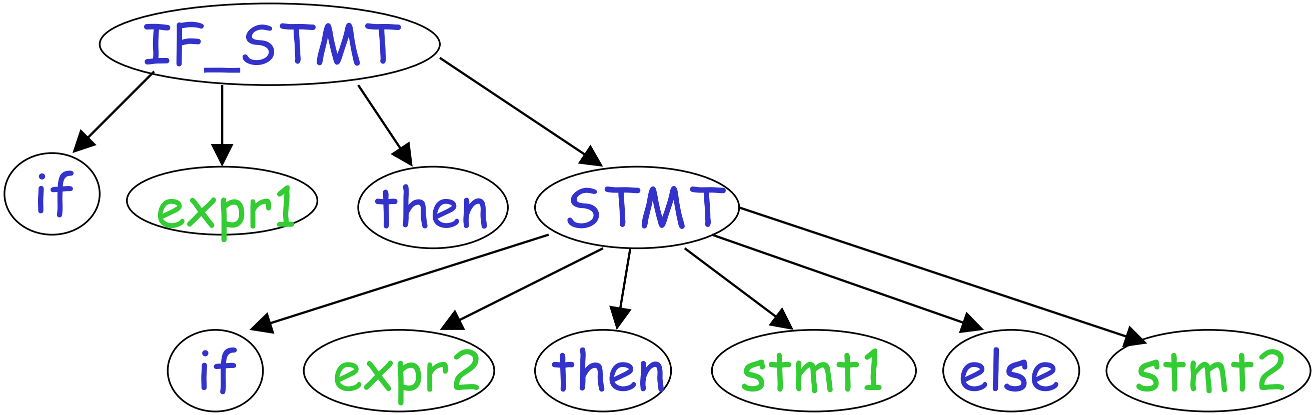


Variables

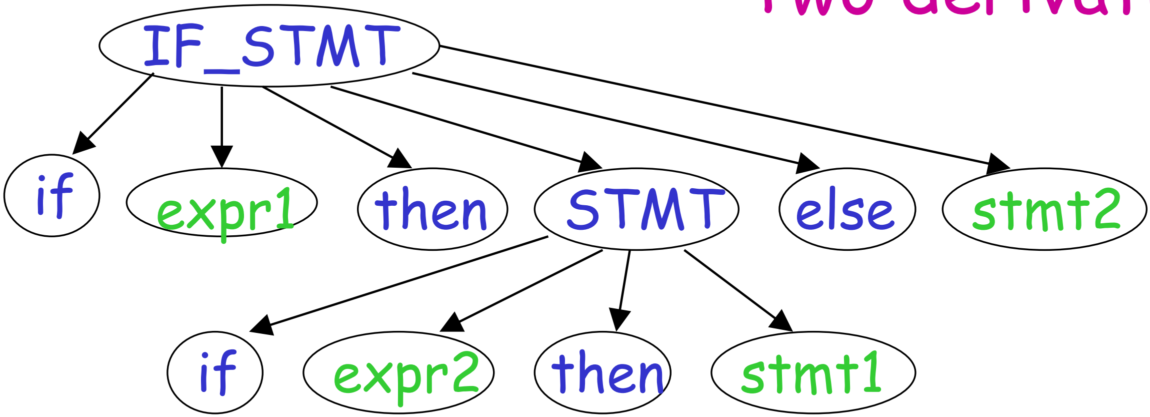
Terminals

Very common piece of grammar
in programming languages

If expr1 then if expr2 then stmt1 else stmt2



Two derivation trees



In general, ambiguity is bad
and we want to remove it

Sometimes it is possible to find
a non-ambiguous grammar for a language

But, in general we cannot do so

A successful example:

Ambiguous Grammar

$$\begin{array}{ll}
E & E \quad E \\
E & E \quad E \\
E & (E\,) \\
E & a
\end{array}$$

Equivalent

Non-Ambiguous Grammar

$$\begin{array}{ll}
E & E \quad T \mid T \\
T & T \quad F \mid F \\
F & (E\,) \mid a
\end{array}$$

generates the same language

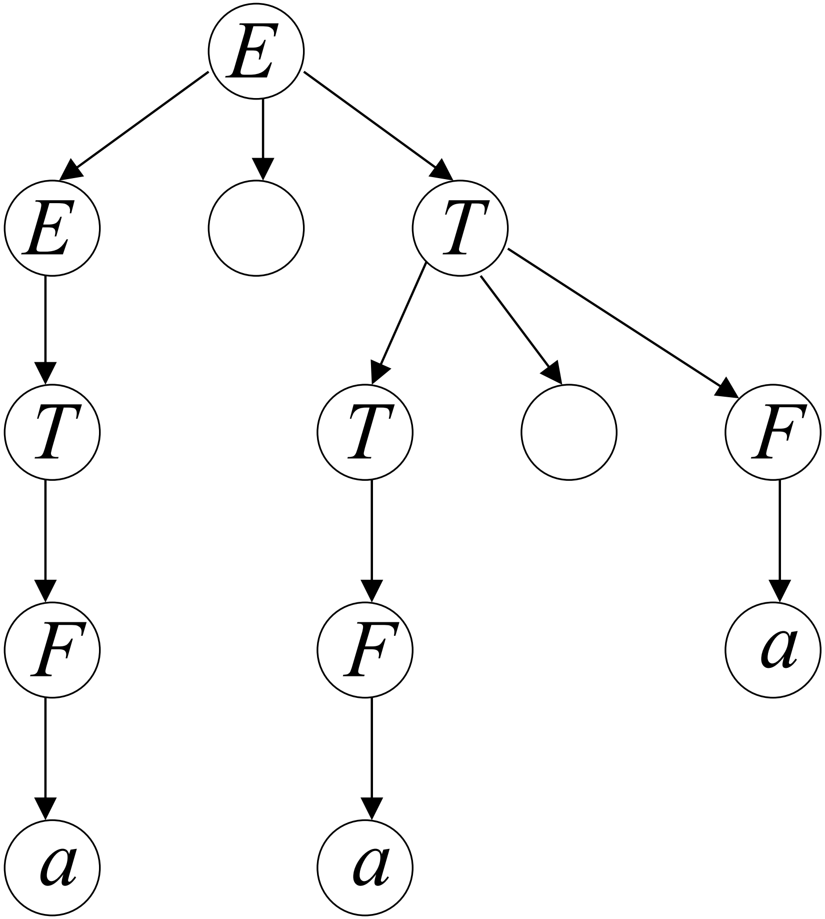
$E \quad E \; T \quad T \; T \quad F \; T \quad a \; T \quad a \; T \; F$

$\quad a \; F \; F \quad a \; a \; F \quad a \; a \; a$

$E \quad E \quad T \,|\, T$

$T \quad T \quad F \,|\, F$

$F \quad (E) \,|\, a$



Unique derivation tree for $a \quad a \quad a$

An un-successful example:

$$L \quad \{a^n b^n c^m\} \qquad \{a^n b^m c^m\}$$

$$n,m \quad 0$$

L is inherently ambiguous:

every grammar that generates this language is ambiguous

Example (ambiguous) grammar for L:

$$L \quad \{a^n b^n c^m\} \qquad \{a^n b^m c^m\}$$

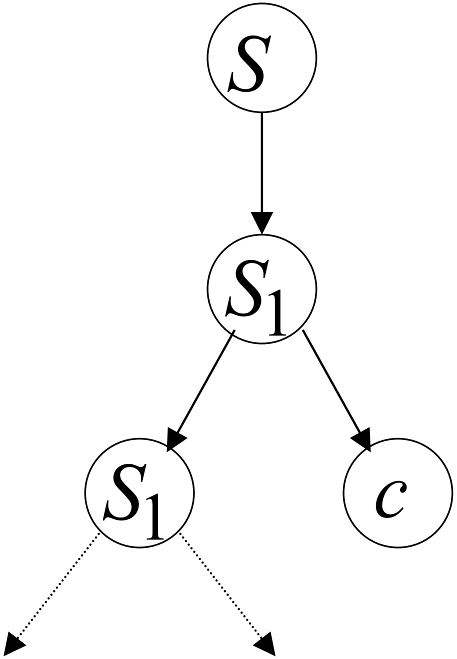
$$S \quad S_1 \mid S_2 \qquad S_1 \quad S_1 c \mid A \qquad S_2 \quad aS_2 \mid B$$

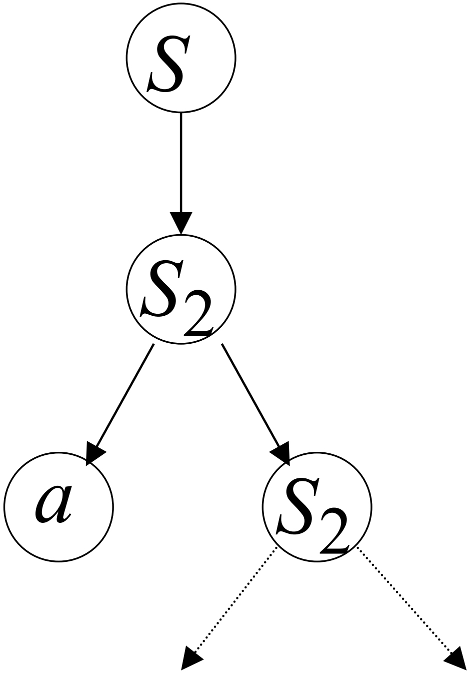
$$A \quad aAb \mid \qquad B \quad bBc \mid$$

The string $a^n b^n c^n \quad L$

has always two different derivation trees (for any grammar)

For example





CFG vs RE

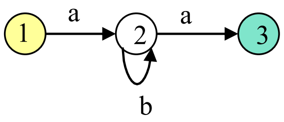
Starting with a NFA:

- For each state S_i in the NFA
  - Create non-terminal A_i
  - If transition $(S_i, a) = S_k$, create production $A_i \to aA_k$
  - If transition $(S_i, \varepsilon) = S_k$, create production $A_i \to A_k$
  - If S_i is a final state, create production $A_i \to \varepsilon$
  - If S_i is the NFA start state, $s = A_i$

- What does the existence of this algorithm tell us about the relationship between regular and context free languages?

NFA to CFG Example

ab*a



$A_1 \rightarrow a\ A_2$

$A_2 \rightarrow b\ A_2$

$A_2 \rightarrow a\ A_3$

$A_3 \rightarrow \varepsilon$

Note:

$\mathbf{S = A_1}$

where S is the start symbol in the CFG.

Writing Grammars

When writing a grammar (or RE) for some language, the following must be true:

1. All strings generated are in the language.
2. Your grammar produces all strings in the language.