Parse Tree

Lin Chen

Email: Lin.Chen@ttu.edu

Grader: zulfi.khan@ttu.edu

TEXAS TECH
UNIVERSITY

Parse tree

- A parse tree is a graphical representation of a derivation

Parse tree

- A parse tree is a graphical representation of a derivation
  - Example:

$$S \rightarrow AB$$
$$A \rightarrow aA|e$$
$$B \rightarrow bB|e$$

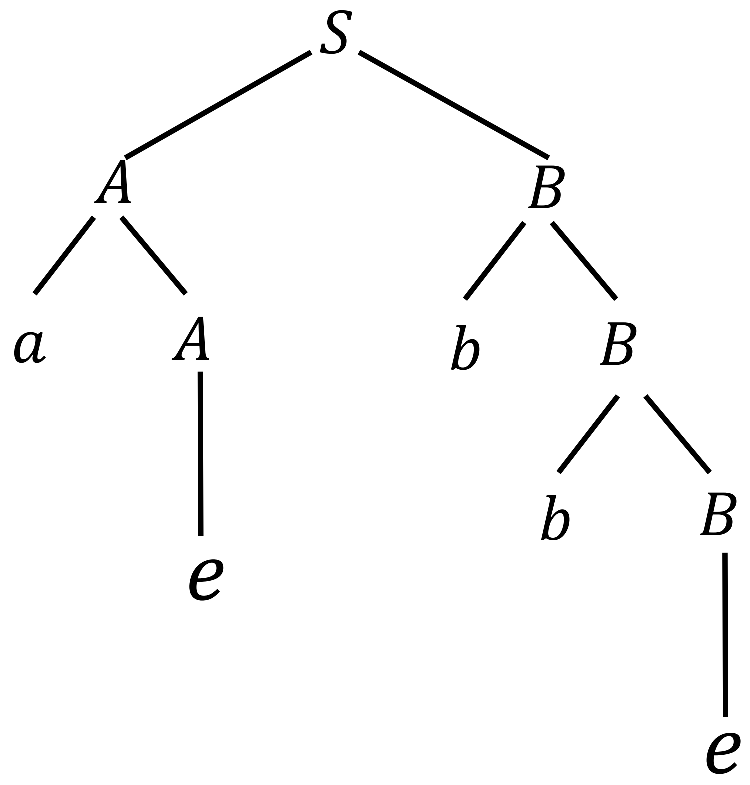
$$\begin{aligned} S &\Rightarrow AB \\ &\Rightarrow aAB \\ &\Rightarrow aAbB \\ &\Rightarrow abB \\ &\Rightarrow abbB \\ &\Rightarrow abb \end{aligned}$$

Parse tree

- A parse tree is a graphical representation of a derivation
  - Example:

$$S \rightarrow AB$$
$$A \rightarrow aA|e$$
$$B \rightarrow bB|e$$

$$\begin{aligned} S &\Rightarrow AB \\ &\Rightarrow aAB \\ &\Rightarrow aAbB \\ &\Rightarrow abB \\ &\Rightarrow abbB \\ &\Rightarrow abb \end{aligned}$$

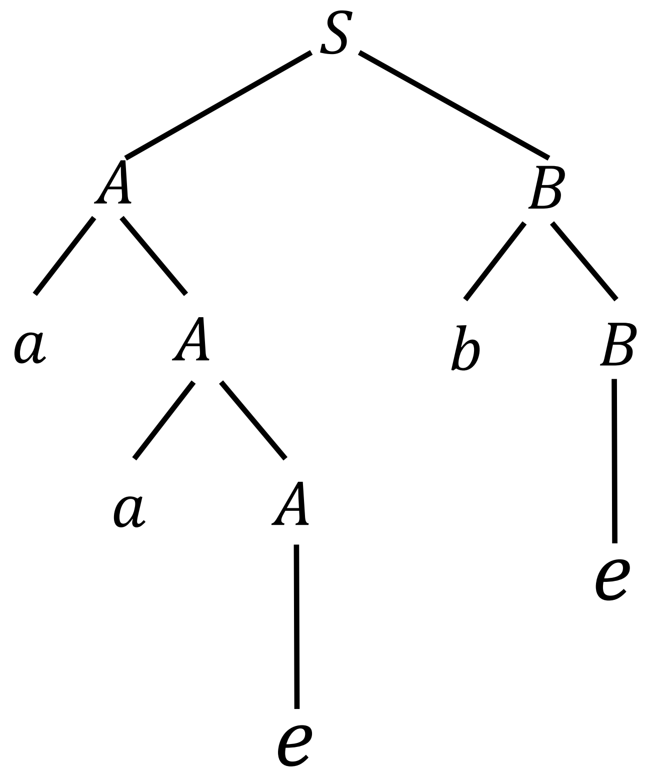


Parse tree

- A parse tree is a graphical representation of a derivation
  - Example:

$$S \to AB$$
$$A \to aA|e$$
$$B \to bB|e$$

$$\begin{aligned} S &\Rightarrow AB \\ &\Rightarrow AbB \\ &\Rightarrow aAbB \\ &\Rightarrow aaAbB \\ &\Rightarrow aaAb \\ &\Rightarrow aab \end{aligned}$$

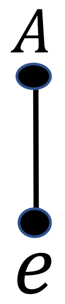


Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - A single node for $a \in \Sigma$ is a parse tree

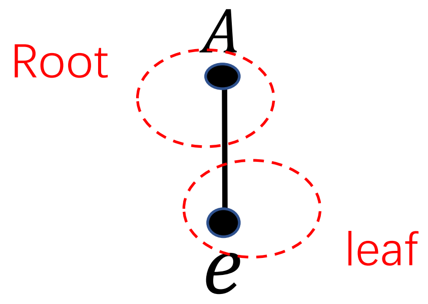
Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - A single node for $a \in \Sigma$ is a parse tree
  - If $A \rightarrow e$ is a rule in R, then the following is a parse tree



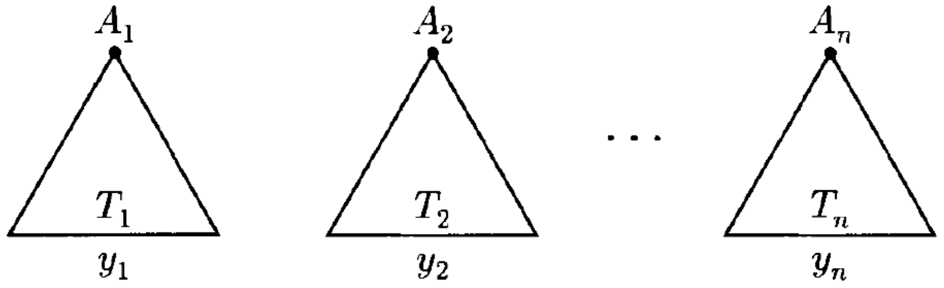
Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - A single node for $a \in \Sigma$ is a parse tree
  - If $A \rightarrow e$ is a rule in R, then the following is a parse tree



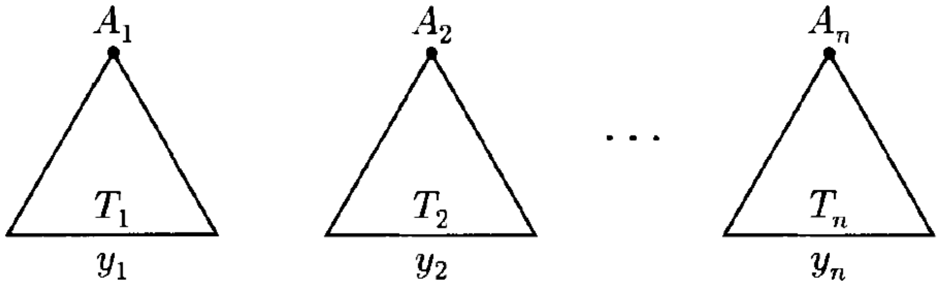
Parse tree

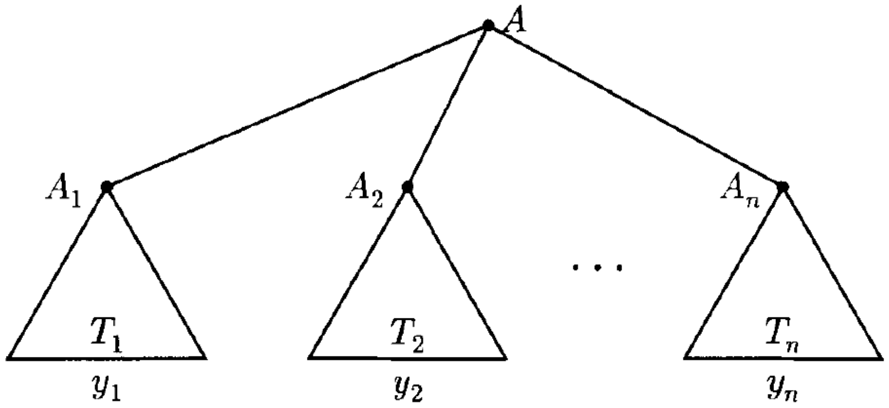
- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - A single node for $a \in \Sigma$ is a parse tree
  - If $A \rightarrow e$ is a rule in R, then the following is a parse tree
  - If the left-belows are parse trees with roots A_i and yield y_i, and $A \rightarrow A_1 \cdots A_n$ is a rule in R



Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - A single node for $a \in \Sigma$ is a parse tree
  - If $A \rightarrow e$ is a rule in R, then the following is a parse tree
  - If the left-belows are parse trees with roots A_i and yield y_i, and $A \rightarrow A_1 \cdots A_n$ is a rule in R, then the right-below is also a parse tree





Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - A single node for $a \in \Sigma$ is a parse tree
  - If $A \rightarrow e$ is a rule in R, then the following is a parse tree
  - If the left-belows are parse trees with roots A_i and yield y_i, and $A \rightarrow A_1 \cdots A_n$ is a rule in R, then the right-below is also a parse tree
  - Nothing else is a parse tree

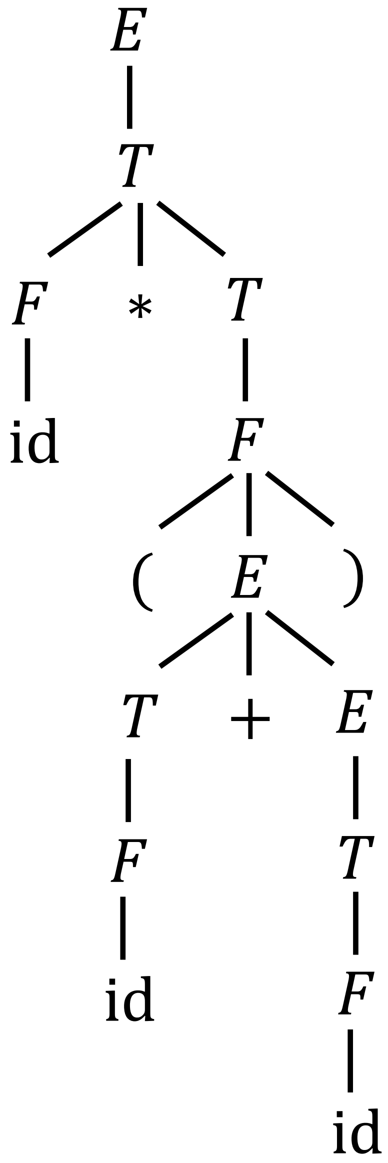
Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - More examples:

$$V = \{+, *, (,), \mathrm{id}, T, F, E\}$$

$$\Sigma = \{+, *, (,), \mathrm{id}\}$$

$$\begin{aligned} E &\to T + E \\ E &\to T \\ T &\to F * T \\ T &\to F \\ F &\to (E) \\ F &\to \mathrm{id} \end{aligned}$$



Parse tree

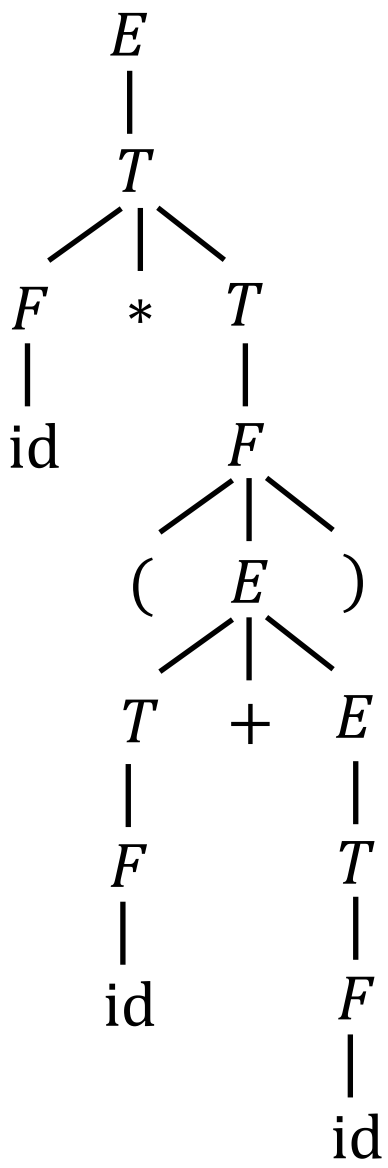
- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - More examples:

$$V = \{+, *, (,), \mathrm{id}, T, F, E\}$$

$$\Sigma = \{+, *, (,), \mathrm{id}\}$$

$$\begin{aligned} E &\to T + E \\ E &\to T \\ T &\to F * T \\ T &\to F \\ F &\to (E) \\ F &\to \mathrm{id} \end{aligned}$$

$\mathrm{id} * (\mathrm{id} + \mathrm{id})$



Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - More examples:

$$
\begin{aligned}
V &= \{S, (,)\},\\
\Sigma &= \{(,)\},\\
R &= \{S \to e,\\
&\qquad S \to SS,\\
&\qquad S \to (S)\}.
\end{aligned}
$$

$$S \Rightarrow SS \Rightarrow (S)S \Rightarrow ()S \Rightarrow ()(S) \Rightarrow ()()$$

$$S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow (S)() \Rightarrow ()()$$

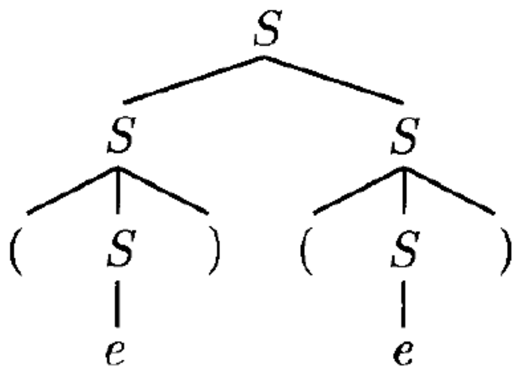
Parse tree

- A parse tree is a graphical representation of a derivation
- Formally, we define a parse tree in an inductive way:
  - More examples:

$$
\begin{aligned}
V &= \{S, (,)\},\\
\Sigma &= \{(,)\},\\
R &= \{S \to e,\\
&\qquad S \to SS,\\
&\qquad S \to (S)\}.
\end{aligned}
$$

$$S \Rightarrow SS \Rightarrow (S)S \Rightarrow ()S \Rightarrow ()(S) \Rightarrow ()()$$

$$S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow (S)() \Rightarrow ()()$$



Parse tree

- Let $G = (V, \Sigma, R, S)$ be a context-free grammar, and let

$$D \Rightarrow x_1 \Rightarrow x_2 \Rightarrow \cdots \Rightarrow x_n$$

$$D' \Rightarrow x_1{}' \Rightarrow x_2{}' \Rightarrow \cdots \Rightarrow x_n{}'$$

$\in V - \Sigma$ $\in \Sigma^*$

We say D precedes D', $D \prec D'$, if $n > 2$ and there is some integer $1 < k < n$ such that

- 1. $x_i = x_i{}'$ for $i \neq k$
- 2. $x_{k-1} = x_{k-1} = uAvBw$, $u, v, w \in V^*$, $A, B \in V - \Sigma$
- 3. $x_k = uyvBw$, where $A \to y \in R$
- 4. $x_k' = uAvzw$, where $B \to z \in R$
- 5. $x_{k+1} = x_{k+1}' = uyvzw$

Parse tree

- Let $G = (V, \Sigma, R, S)$ be a context-free grammar, and let

$$D \Rightarrow x_1 \Rightarrow x_2 \Rightarrow \cdots \Rightarrow x_n$$

$$D' \Rightarrow x_1{}' \Rightarrow x_2{}' \Rightarrow \cdots \Rightarrow x_n{}'$$

$\in V - \Sigma$ $\in \Sigma^*$

We say D precedes D', $D \prec D'$, if $n > 2$ and there is some integer $1 < k < n$ such that

- 1. $x_i = x_i{}'$ for $i \neq k$
- 2. $x_{k-1} = x_{k-1} = uAvBw$, $u, v, w \in V^*$, $A, B \in V - \Sigma$
- 3. $x_k = uyvBw$, where $A \to y \in R$
- 4. $x_k' = uAvzw$, where $B \to z \in R$
- 5. $x_{k+1} = x_{k+1}' = uyvzw$

Differs in the order of only one step

Parse tree

- Example:

$$
\begin{aligned}
V &= \{S, (,)\},\\
\Sigma &= \{(,)\},\\
R &= \{S \to e,\\
&\qquad S \to SS,\\
&\qquad S \to (S)\}.
\end{aligned}
$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$

$$D_2 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$

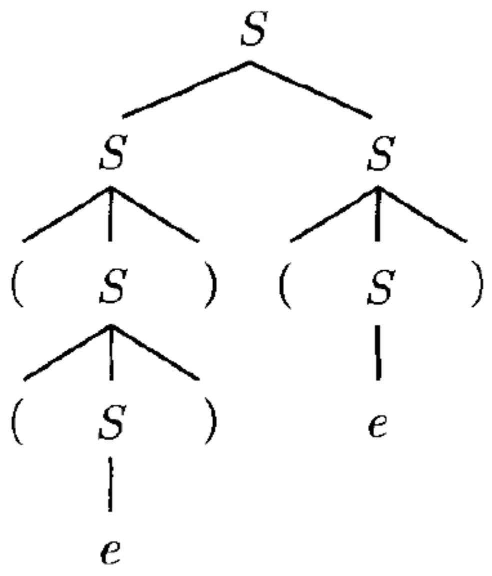
$$D_3 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$

Parse tree

- Example:

$$
\begin{aligned}
V &= \{S, (,)\},\\
\Sigma &= \{(,)\},\\
R &= \{S \to e,\\
&\qquad S \to SS,\\
&\qquad S \to (S)\}.
\end{aligned}
$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$
$$D_2 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_3 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$



Parse tree

- Example:

$$V = \{S, (,)\},$$
$$\Sigma = \{(,)\},$$
$$R = \{S \to e,$$
$$S \to SS,$$
$$S \to (S)\}.$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$
$$D_2 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_3 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$

$$D_1 \prec D_2$$

Parse tree

- Example:

$$V = \{S, (,)\},$$
$$\Sigma = \{(,)\},$$
$$R = \{S \to e,$$
$$S \to SS,$$
$$S \to (S)\}.$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$
$$D_2 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_3 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$

$$D_1 \prec D_2$$
$$D_2 \prec D_3$$

Parse tree

- Example:

$$V = \{S, (,)\},$$
$$\Sigma = \{(,)\},$$
$$R = \{S \to e,$$
$$S \to SS,$$
$$S \to (S)\}.$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$
$$D_2 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_3 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$

$$D_1 \prec D_2$$
$$D_2 \prec D_3$$

D, D' are similar if $D \prec \cdots \prec D'$ or $D' \prec \cdots \prec D$

- D_1, D_2, D_3 are similar

Parse tree

- Example:

$$V = \{S, (,)\},$$
$$\Sigma = \{(,)\},$$
$$R = \{S \to e,$$
$$S \to SS,$$
$$S \to (S)\}.$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$
$$D_2 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_3 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$
$$D_4 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_5 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$
$$D_6 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)(S) \Rightarrow (S)() \Rightarrow ((S))() \Rightarrow (())()$$
$$D_7 = S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())()$$
$$D_8 = S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())()$$
$$D_9 = S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow (S)() \Rightarrow ((S))() \Rightarrow (())()$$
$$D_{10} = S \Rightarrow SS \Rightarrow S(S) \Rightarrow S() \Rightarrow (S)() \Rightarrow ((S))() \Rightarrow (())()$$

Parse tree

- Example:

$$
\begin{aligned}
V &= \{S, (,)\}, \\
\Sigma &= \{(,)\}, \\
R &= \{S \to e, \\
&\qquad S \to SS, \\
&\qquad S \to (S)\}.
\end{aligned}
$$

$$
\begin{aligned}
D_1 &= S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())() \\
D_2 &= S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())() \\
D_3 &= S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())() \\
D_4 &= S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())() \\
D_5 &= S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())() \\
D_6 &= S \Rightarrow SS \Rightarrow (S)S \Rightarrow (S)(S) \Rightarrow (S)() \Rightarrow ((S))() \Rightarrow (())() \\
D_7 &= S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow (())(S) \Rightarrow (())() \\
D_8 &= S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow ((S))(S) \Rightarrow ((S))() \Rightarrow (())() \\
D_9 &= S \Rightarrow SS \Rightarrow S(S) \Rightarrow (S)(S) \Rightarrow (S)() \Rightarrow ((S))() \Rightarrow (())() \\
D_{10} &= S \Rightarrow SS \Rightarrow S(S) \Rightarrow S() \Rightarrow (S)() \Rightarrow ((S))() \Rightarrow (())()
\end{aligned}
$$

$$
\begin{tikzcd}
 & & D_3 \arrow[dr, phantom, "\prec"] & & & & \\
D_1 \arrow[r, phantom, "\prec"] & D_2 \arrow[ur, phantom, "\prec"] \arrow[dr, phantom, "\prec"] & & D_5 \arrow[r, phantom, "\prec"] & D_6 \arrow[dr, phantom, "\prec"] & & \\
 & & D_4 \arrow[ur, phantom, "\prec"] \arrow[dr, phantom, "\prec"] & & & D_9 \arrow[r, phantom, "\prec"] & D_{10} \\
 & & & D_7 \arrow[r, phantom, "\prec"] & D_8 \arrow[ur, phantom, "\prec"] & &
\end{tikzcd}
$$

Parse tree

- Similarity is an equivalence relation (reflexive, symmetric, transitive)
- Similar derivations have the same parse tree

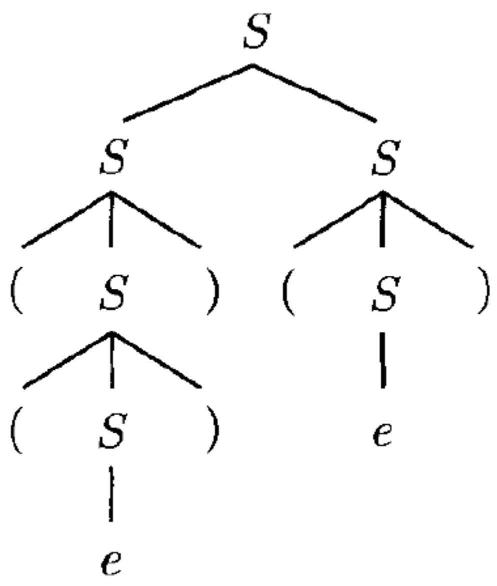
Parse tree

- Similarity is an equivalence relation (reflexive, symmetric, transitive)
- Similar derivations have the same parse tree
- Similar derivations generate the same string at last
  - Is the opposite true?

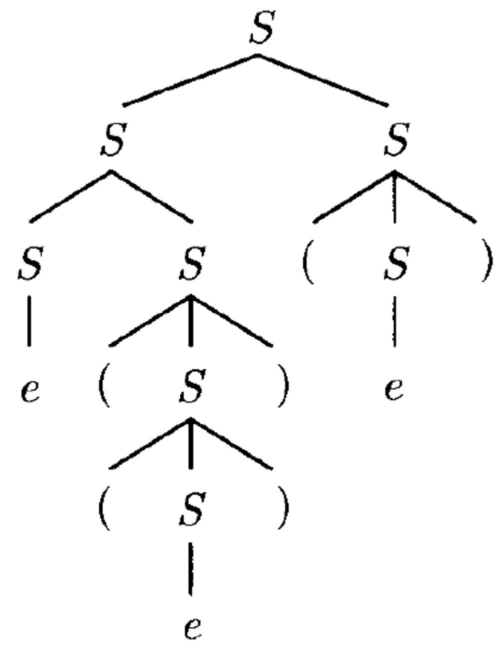
Parse tree

- Similar derivations generate the same string at last
  - Is the opposite true?

$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S$
$\Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$



$D' = S \Rightarrow SS \Rightarrow SSS \Rightarrow S(S)S \Rightarrow S((S))S$
$\Rightarrow S(())S \Rightarrow S(())(S) \Rightarrow S(())() \Rightarrow (())()$



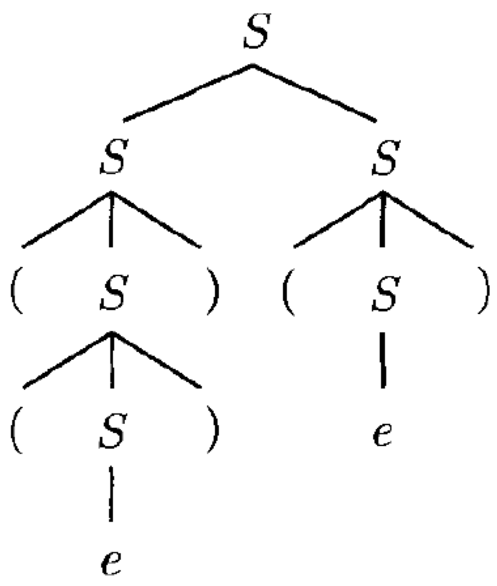
Ambiguity

- Same parse tree yields same string
  - Is the opposite true?

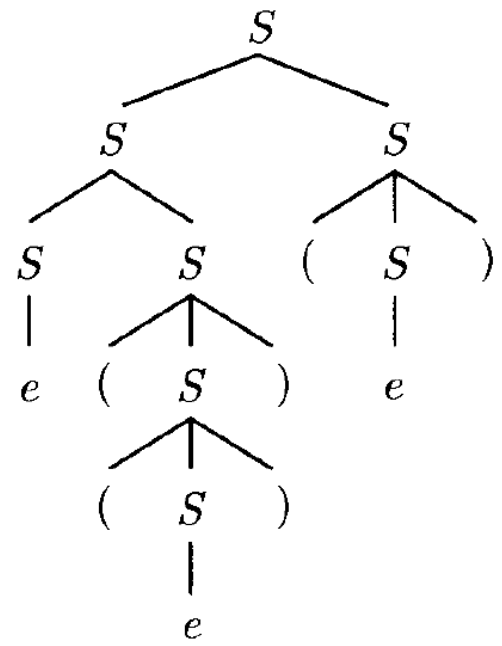
Ambiguity

- Same parse tree yields same string
  - Is the opposite true?

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow ((S))S \Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$



$$D' = S \Rightarrow SS \Rightarrow SSS \Rightarrow S(S)S \Rightarrow S((S))S \Rightarrow S(())S \Rightarrow S(())(S) \Rightarrow S(())() \Rightarrow (())()$$



Ambiguity

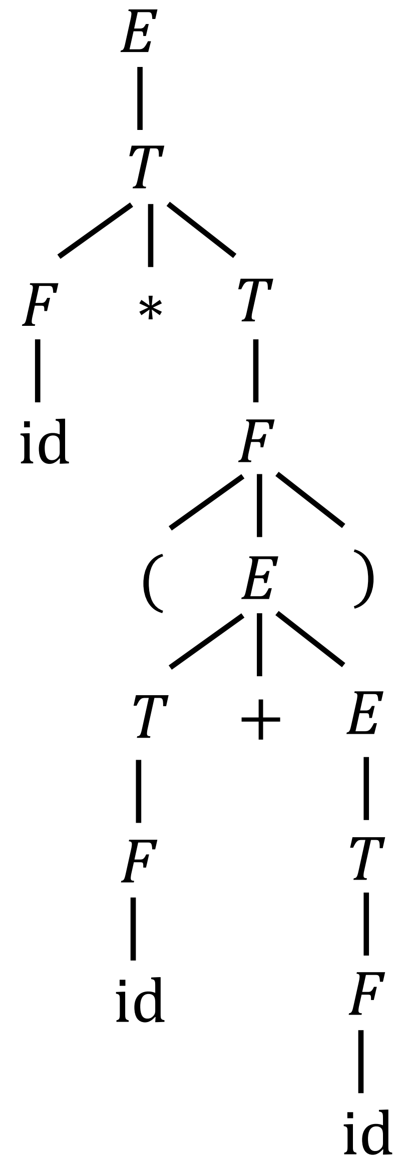
- Same parse tree yields same string
  - Is the opposite true?

$$V = \{+, *, (,), \mathrm{id}, T, F, E\}$$

$$\Sigma = \{+, *, (,), \mathrm{id}\}$$

$$E \to T + E$$
$$E \to T$$
$$T \to F * T$$
$$T \to F$$
$$F \to (E)$$
$$F \to \mathrm{id}$$

$\mathrm{id} * (\mathrm{id} + \mathrm{id})$



Ambiguity

- Same parse tree yields same string
  - Is the opposite true?

$V = \{+, *, (,), \mathrm{id}, T, F, E\}$

$\Sigma = \{+, *, (,), \mathrm{id}\}$

$E \to T + E$ $\quad\quad$ $E \to E + E$

$\cancel{E \to T}$

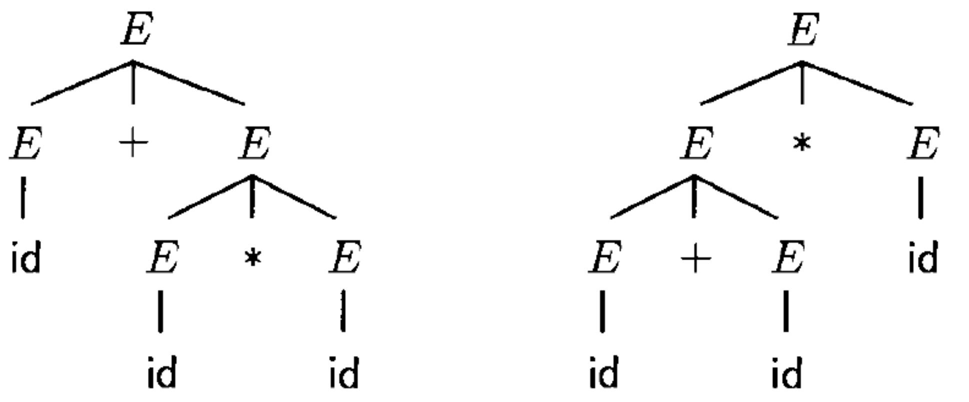
$T \to F * T$ $\quad\quad$ $E \to E * E$

$\cancel{T \to F}$

$F \to (E)$ $\quad\quad$ $E \to (E)$

$F \to \mathrm{id}$ $\quad\quad$ $E \to \mathrm{id}$

We can still generate $\mathbf{id} * (\mathbf{id} + \mathbf{id})$, but⋯



(a) (b)

Ambiguity

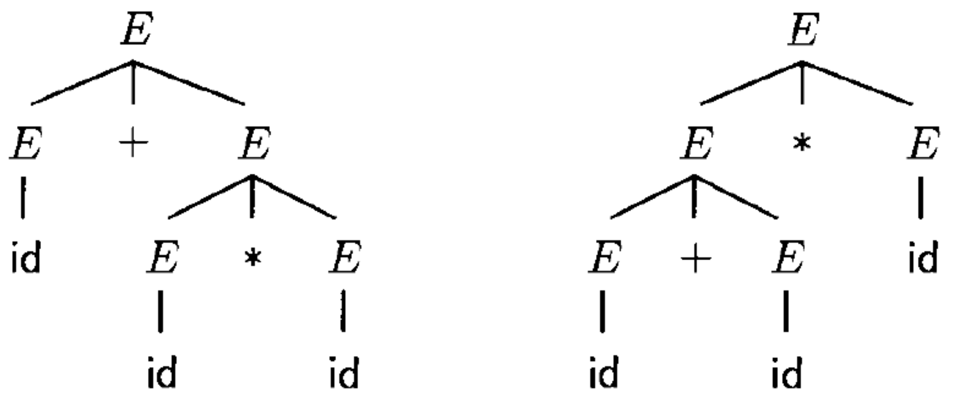
- A Grammar with a string that has two or more distinct parse trees is called ambiguous

$V = \{+, *, (,), \mathrm{id}, T, F, E\}$

$\Sigma = \{+, *, (,), \mathrm{id}\}$

$E \rightarrow T + E$	$E \rightarrow E + E$
$\cancel{E \rightarrow T}$	
$T \rightarrow F * T$	$E \rightarrow E * E$
$\cancel{T \rightarrow F}$	
$F \rightarrow (E)$	$E \rightarrow (E)$
$F \rightarrow \mathrm{id}$	$E \rightarrow \mathrm{id}$

We can still generate $\mathbf{id} * (\mathbf{id} + \mathbf{id})$, but⋯



(a) (b)

Ambiguity

- A Grammar with a string that has two or more distinct parse trees is called ambiguous
- Can we construct an unambiguous grammar?

$$V = \{S, (,)\},$$
$$\Sigma = \{(,)\},$$
$$R = \{S \to e,$$
$$S \to SS,$$
$$S \to (S)\}.$$

$$D_1 = S \Rightarrow SS \Rightarrow (S)S \Rightarrow \big((S)\big)S$$
$$\Rightarrow (())S \Rightarrow (())(S) \Rightarrow (())()$$

$$D' = S \Rightarrow SS \Rightarrow SSS \Rightarrow S(S)S \Rightarrow S\big((S)\big)S$$
$$\Rightarrow S(())S \Rightarrow S(())(S) \Rightarrow S(())() \Rightarrow (())()$$

Ambiguity

- A Grammar with a string that has two or more distinct parse trees is called ambiguous
- Can we construct an unambiguous grammar?

$$
\begin{aligned}
V &= \{S, (,)\},\\
\Sigma &= \{(,)\},\\
R &= \{S \to e,\\
&\quad\; S \to AS,\\
&\quad\; A \to (S)\}.
\end{aligned}
$$

Ambiguity

- A Grammar with a string that has two or more distinct parse trees is called ambiguous
- There exist context-free languages such that all context-free grammars that generate them must be ambiguous (inherently ambiguous)
- A programming language should not be ambiguous