COMP3173 Compiler Construction
Top-Down Parsing

Dr. Zhiyuan Li

Outline

- Nonrecursive Parser
- First
- Follow
- Parsing Table Construction

Nonrecursive Predictive Parsers

- To avoid recursions, we introduce nonrecursive predictive parsers.
- Intuitively, predictions are required when a nonterminal has multiple production rules.
- The predictions are based on the next token. One token is enough because we are parsing $LL(1)$ grammars.
- However, this method has troubles when the RHS of some rules start with nonterminals.
- For example,

$$\begin{aligned} A &\rightarrow B|C \\ B &\rightarrow b \\ C &\rightarrow c \end{aligned}$$

we cannot decide whether we are going to use $A \rightarrow B$ or $A \rightarrow C$ without looking at $B \rightarrow b$ and $C \rightarrow c$.

Nonrecursive Predictive Parsers

- Thus, we can "preprocess" the grammar to find the first tokens derived by each nonterminal before parsing the tokens.
- The result of preprocessing is presented by a table, called ***parsing table***.
- The rows are the nonterminals and the columns are terminals.
- The entry $M[A, a]$ is a production rule $A \rightarrow \alpha$, meaning that if the input token is a, we will apply the rule $A \rightarrow \alpha$.
- The parser can parse the input tokens by checking the parsing table and using a stack.
- This method is similar to the push-down automata but presented differently.

Parsing using a parsing table

- Initially, push $ and E into the stack (E on top of $) where E is the first nonterminal. Also insert $ to the end of the input stream of tokens.
- In each iteration,
  - assume the next input token is a;
  - pop the first item from the stack denoted by X;
  - If X is a terminal, then try to match it with the next input token a;
  - If X is a nonterminal, then $M[X, a]$ in the parsing table is a production rule, denoted by $X \rightarrow \alpha$. Then, push everything in α to stack from right to left.
- When the stack pops $ and all input tokens are consumed, the parsing halts.
- $ is an artificial token, means the end of the stream.

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
	$(id + id) * id\$$	$\$E$	

- To save space, the stack is represented in a line. The right most symbol is the top of the stack.

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$E \to TE'$	$(id + id) * id\$$	$\$E'T$	E

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T \to FT'$	$(id + id) * id\$$	$\$E'T'F$	T

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$F \to (E)$	$(id + id) * id\$$	$\$E'T')E($	F

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$(id + id) * id\$$	$\$E'T')E$	$($

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$E \to TE'$	$id + id) * id\$$	$\$E'T')E'T$	E

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T \to FT'$	$id + id) * id\$$	$\$E'T')E'T'F$	T

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$F \to id$	$id + id) * id\$$	$\$E'T')E'T'id$	F

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$id + id) * id\$$	$\$E'T')E'T'$	id

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T' \to \varepsilon$	$+id) * id\$$	$\$E'T')E'$	T'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$E' \to +TE'$	$+id) * id\$$	$\$E'T')E'T +$	E'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$+id) * id\$$	$\$E'T')E'T$	$+$

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T \to FT'$	$id) * id\$$	$\$E'T')E'T'F$	T

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$F \to id$	$id) * id\$$	$\$E'T')E'T'id$	F

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$id) * id\$$	$\$E'T')E'T'$	id

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T' \to \varepsilon$	$) * id\$$	$\$E'T')E'$	T'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$E' \to \varepsilon$	$) * id\$$	$\$E'T')$	E'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$) * id\$$	$\$E'T'$	$)$

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T' \to * FT'$	$* id\$$	$\$E'T'F *$	T'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$* id\$$	$\$E'T'F$	$*$

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$F \to id$	$id\$$	$\$E'T'id$	F

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$id\$$	$\$E'T'$	id

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$T' \to \varepsilon$	$	$\$E'$	T'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$E' \to \varepsilon$	$	$	E'

Example

- Given the parsing table and try to parse
$$(id + id) * id$$

	id	$+$	$*$	$($	$)$	$\$$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \varepsilon$	$E' \to \varepsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \varepsilon$	$T' \to * FT'$		$T' \to \varepsilon$	$T' \to \varepsilon$
F	$F \to id$			$F \to (E)$		

Production Rule	Input	Stack	Pop
$Token\ Matched$	$		$

Nonrecursive Predictive Parsers

- Some entries $M[A, a]$ in the parsing table are empty.
- Meaning that if you try to parse the token a by a nonterminal A, the parser returns an error message.
- For example, assume the input tokens are $id + * idid$, the parsing will be stuck somewhere in the middle.
- The error handling will be discussed at the end.
- Intuitively, a parsing table enumerates all possible tokens can be derived by a nonterminal.
- When a parser reads a token from input, it has a unique production rule to apply.
- Thus, we need to analyze the grammar and find the prefix (the first token) of all possible sentential form derived by each nonterminal.

First and Follow

- By putting all the firstly produced tokens into a set, we defined $First()$.
- For example,

$$E \rightarrow E + T \mid E - T \mid T$$
$$T \rightarrow id \mid (E)$$

- $id \in First(T)$ because $T \Rightarrow id$.
- $(\in First(T)$ because $T \Rightarrow (E)$.
- $id \in First(E)$ because $E \Rightarrow T \Rightarrow id$.
- $(\in First(E)$ because $E \Rightarrow T \Rightarrow (E)$.

First and Follow

- Formally, $First(X)$ for each grammar symbol is computed by

Algorithm: Find_First(G)

Input: an $LL(1)$ grammar G

Output: $First(X)$ for every grammar symbol X

1. **If** X is a token, then $First(X) = \{X\}$.
2. **If** X is a nonterminal and $X \to \varepsilon$, then add ε to $First(X)$.
3. **If** X is a nonterminal and $X \to Y_1 Y_2 \cdots Y_k$, then add $First(Y_1) \setminus \{\varepsilon\}$ to $First(X)$.
4. **If** X is a nonterminal and $X \to Y_1 Y_2 \cdots Y_k$ and $\varepsilon \in First(Y_1) \cdots First(Y_i)$, then add $First(Y_{i+1}) \setminus \{\varepsilon\}$ to $First(X)$.
5. **If** X is a nonterminal and $X \to Y_1 Y_2 \cdots Y_k$ and $\varepsilon \in First(Y_i)$ for all $1 \leq i \leq k$, then add ε to $First(X)$.
6. Repeat 1-5 until no more changes.

- Note: each Y_i on line 3-5 can be either a terminal or a nonterminal.

Example

- Given the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' | \varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT' | \varepsilon \\
F &\rightarrow (E) | id
\end{aligned}
$$

- Applying Rule 1 "**If** X is a token, then $First(X) = \{X\}$."

X	***id***	+	*	(	)
$First(X)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{($ $\}$	$\{)\}$

- Applying Rule 2 "**If** X is a nonterminal and $X \rightarrow \varepsilon$, then add ε to $First(X)$."

X	***E***	***E′***	***T***	***T′***	***F***
$First(X)$		$\{\varepsilon\}$		$\{\varepsilon\}$	

Example

- Given the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' | \varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT' | \varepsilon \\
F &\rightarrow (E) | id
\end{aligned}
$$

- Applying Rule 3 "**If** X is a nonterminal and $X \rightarrow Y_1 Y_2 \cdots Y_k$, then add $First(Y_1) \setminus \{\varepsilon\}$ to $First(X)$."

X	***id***	+	*	(	)
$First(X)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{($ $\}$	$\{)\}$

X	***E***	***E′***	***T***	***T′***	***F***
$First(X)$		$\{\varepsilon, +\}$		$\{\varepsilon, *\}$	$\{(, id\}$

Example

- Given the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' | \varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT' | \varepsilon \\
F &\rightarrow (E) | id
\end{aligned}
$$

- Applying Rule 3 again

X	id	+	*	(	)
$First(X)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{($ $\}$	$\{)\}$

X	E	E'	T	T'	F
$First(X)$		$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

Example

- Given the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' | \varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT' | \varepsilon \\
F &\rightarrow (E) | id
\end{aligned}
$$

- Applying Rule 3 again

X	***id***	+	*	(	)
$First(X)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{($ $\}$	$\{)\}$

X	***E***	***E′***	***T***	***T′***	***F***
$First(X)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

Exercise

- Find $First$ sets for the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' \mid -TE' \mid \varepsilon \\
T &\rightarrow id \mid (E)
\end{aligned}
$$

First and Follow

- When a terminal produces ε, things get complicated. For example,

$$\begin{aligned} A &\rightarrow BC \\ B &\rightarrow bB|\varepsilon \\ C &\rightarrow c \end{aligned}$$

- The parser need to use the rule $B \rightarrow \varepsilon$ to parse c.

$$A \Rightarrow BC \Rightarrow \varepsilon C \Rightarrow \varepsilon c$$

- However, ε is not an input token which can be used for making predictions.
- To handle this case, we also define $Follow(E)$, the next token possibly derived by some rules after E.
- In this example, $c \in Follow(B)$ because $A \Rightarrow BC \Rightarrow Bc$.
- When the parser receives c as the next token, it knows $B \rightarrow \varepsilon$ is the correct rule.

First and Follow

- And $Follow(X)$ is computed by

Algorithm: Find_Follow(G)

Input: an $LL(1)$ grammar G

Output: $Follow(X)$ for every grammar symbol X

1. **If** X is a start symbol, then $Follow(X) = \{\$\}$.
2. **If** there is a production $X \to Y_1 Y_2 \cdots Y_k$, then add $First(Y_i) \setminus \{\varepsilon\}$ to $Follow(Y_{i-1})$ for all $1 < i \leq k$.
3. **If** there is a production $X \to Y_1 Y_2 \cdots Y_k$, then add $Follow(X)$ to $Follow(Y_k)$.
4. **If** there is a production $X \to Y_1 Y_2 \cdots Y_k$ and $\varepsilon \in First(Y_i), \cdots, First(Y_k)$, then add $Follow(X)$ to $Follow(Y_{i-1})$.
5. Repeat 1-4 until no more changes.

Example

- To find $Follow$ of

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow *FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

$First(x)$ for nonterminals

E	E'	T
$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$
T'	F	
$\{\varepsilon, *\}$	$\{(, id\}$	

- Applying Rule 1:

If X is a start symbol, then $Follow(X) = \{\$\}$.

X	E	E'	T	T'	F
$Follow(X)$	{$}	{}	{}	{}	{}

Example

- To find $Follow$ of

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow *FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

$First(x)$ for nonterminals

E	E'	T
$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$
T'	F	
$\{\varepsilon, *\}$	$\{(, id\}$	

- Applying Rule 2

If there is a production $X \rightarrow Y_1 Y_2 \cdots Y_k$, then add $First(Y_2) \setminus \{\varepsilon\}$ to $Follow(Y_1)$.

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\}$	$\{+\}$	$\{\}$	$\{*\}$

Example

- To find $Follow$ of

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow *FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

$First(x)$ for nonterminals

E	E'	T
$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$
T'	F	
$\{\varepsilon, *\}$	$\{(, id\}$	

- Applying Rule 3

If there is a production $X \rightarrow Y_1 Y_2 \cdots Y_k$, then add $Follow(X)$ to $Follow(Y_k)$.

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+\}$	$\{+\}$	$\{*\}$

Example

- To find $Follow$ of

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow *FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

$First(x)$ for nonterminals

E	E'	T
$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$
T'	F	
$\{\varepsilon, *\}$	$\{(, id\}$	

- Applying Rule 4

If there is a production $X \rightarrow Y_1 Y_2 \cdots Y_k$ and $\varepsilon \in First(Y_i), \cdots, First(Y_k)$, then add $Follow(X)$ to $Follow(Y_{i-1})$.

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+\}$	$\{*, +\}$

Example

- To find $Follow$ of

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow *FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

$First(x)$ for nonterminals

E	E'	T
$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$
T'	F	
$\{\varepsilon, *\}$	$\{(, id\}$	

- Applying Rule 3

If there is a production $X \rightarrow Y_1 Y_2 \cdots Y_k$, then add $Follow(X)$ to $Follow(Y_k)$.

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +\}$

Example

- To find $Follow$ of

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow *FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

$First(x)$ for nonterminals

E	E'	T
$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$
T'	F	
$\{\varepsilon, *\}$	$\{(, id\}$	

- Applying Rule 4

If there is a production $X \rightarrow Y_1 Y_2 \cdots Y_k$ and $\varepsilon \in First(Y_i), \cdots, First(Y_k)$, then add $Follow(X)$ to $Follow(Y_{i-1})$.

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

Exercise

- Find $Follow$ sets for the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' \mid -TE' \mid \varepsilon \\
T &\rightarrow id \mid (E)
\end{aligned}
$$

Construct the parsing table

- Recall the parsing table M is $n \times m$, where n is the number of nonterminals and m is the number of tokens.
- The entry $M[A, a]$ is a production rule that the parser will apply when the current nonterminal is A and the next token is a.
- For each production $A \rightarrow \alpha$ in the (left factored, non-left recursive) grammar, do the following:
  1. For each token a in $First(\alpha)$, add the grammar production $A \rightarrow \alpha$ to $M[A, a]$.
  2. If ε is in $First(\alpha)$ then, for each token b in $Follow(A)$, add $A \rightarrow \alpha$ to $M[A, b]$.
  3. All other entries in the table are left blank and correspond to a syntax error.
- Note that Rule 2 is applied when α is ε because ε is in $First(\varepsilon)$.

Example

$E \rightarrow TE'$

$E' \rightarrow +TE'|\varepsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT'|\varepsilon$

$F \rightarrow (E)|id$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 1, add $E \rightarrow TE'$ to $M[E, id]$ and $M[E, (]$ because $First(T) = \{(, id\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'						
T						
T'						
F						

Example

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' | \varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT' | \varepsilon \\
F &\rightarrow (E) | id
\end{aligned}
$$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 1, add $E' \rightarrow +TE'$ to $M[E, +]$ because $First(+) = \{+\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$				
T						
T'						
F						

Example

$E \rightarrow TE'$

$E' \rightarrow +TE'|\varepsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT'|\varepsilon$

$F \rightarrow (E)|id$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 1, add $T \rightarrow FT'$ to $M[T, id]$ and $M[T, (]$ because $First(F) = \{(, id\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$				
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'						
F						

Example

$E \rightarrow TE'$

$E' \rightarrow +TE'|\varepsilon$

$T \rightarrow FT'$

$T' \rightarrow * FT'|\varepsilon$

$F \rightarrow (E)|id$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 1, add $T' \rightarrow * FT'$ to $M[T', *]$ because $First(*) = \{*\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$				
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'			$T' \rightarrow * FT'$			
F						

Example

$E \rightarrow TE'$

$E' \rightarrow +TE'|\varepsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT'|\varepsilon$

$F \rightarrow (E)|id$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 1, add $F \rightarrow (E)$ to $M[F, (]$ because $First(() = \{(\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$				
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'			$T' \rightarrow *FT'$			
F				$F \rightarrow (E)$		

Example

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 1, add $F \rightarrow id$ to $M[F, id]$ because $First(id) = \{id\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$				
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'			$T' \rightarrow * FT'$			
F	$F \rightarrow id$			$F \rightarrow (E)$		

Example

$E \rightarrow TE'$

$E' \rightarrow +TE' | \varepsilon$

$T \rightarrow FT'$

$T' \rightarrow * FT' | \varepsilon$

$F \rightarrow (E) | id$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 2, add $E' \rightarrow \varepsilon$ to $M[E', \$]$ and $M[E',)]$ because $Follow(E') = \{\$,)\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \varepsilon$	$E' \rightarrow \varepsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'			$T' \rightarrow * FT'$			
F	$F \rightarrow id$			$F \rightarrow (E)$		

Example

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE'|\varepsilon \\
T &\rightarrow FT' \\
T' &\rightarrow * FT'|\varepsilon \\
F &\rightarrow (E)|id
\end{aligned}
$$

X	id	$+$	$*$	$($	$)$
$First(x)$	$\{id\}$	$\{+\}$	$\{*\}$	$\{(\}$	$\{)\}$
X	E	E'	T	T'	F
$First(x)$	$\{(, id\}$	$\{\varepsilon, +\}$	$\{(, id\}$	$\{\varepsilon, *\}$	$\{(, id\}$

X	E	E'	T	T'	F
$Follow(X)$	$\{\$,)\}$	$\{\$,)\}$	$\{+, \$,)\}$	$\{+, \$,)\}$	$\{*, +, \$,)\}$

- Rule 2, add $T' \rightarrow \varepsilon$ to $M[T', +]$, $M[T', \$]$, and $M[T',)]$ because $Follow(T') = \{+, \$,)\}$.

	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \varepsilon$	$E' \rightarrow \varepsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \varepsilon$	$T' \rightarrow * FT'$		$T' \rightarrow \varepsilon$	$T' \rightarrow \varepsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Overview on the nonrecursive parsers

To create a parsing table for a nonrecursive parser,

- eliminate any ambiguity from the grammar,
- eliminate left recursion from the grammar,
- left factor the grammar,
- compute the $First$ sets for all tokens and nonterminals,
- compute the $Follow$ sets for all nonterminals, and
- use those $First$ and $Follow$ sets to construct the parsing table.

Exercise

- Construct the parsing table for the following grammar

$$
\begin{aligned}
E &\rightarrow TE' \\
E' &\rightarrow +TE' \mid -TE' \mid \varepsilon \\
T &\rightarrow id \mid (E)
\end{aligned}
$$

End of lecture 6