# LR Parsing

Dr. Mattox Beckman

University of Illinois at Urbana-Champaign
Department of Computer Science



## Objectives

You should be able to ...

- Explain the difference between an LL and LR parser.
- Generate the finite state machine from an LR grammar.
- Use the state machine to detect ambiguity.

Further reading: See Dragon Book §4.x.



## What Is LR Parsing?

- What is an LR parser?
  - An LR parser uses a **L**eft-to-right scan and produces a **R**ightmost derivation.
  - A.k.a. bottom-up parsing
  - Uses a *push-down automata* to do the work.
- There are four actions.

| | |
|---|---|
| Shift | Consume a token from the input. |
| Reduce | Build a tree from components. |
| Goto | Jump to a different state, after a reduce. |
| Accept | Signal that we're done. |

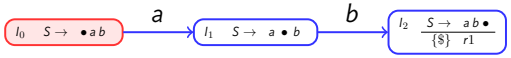


## Shifting

Shifting involves three steps.

1. Consume a token from the input.
2. Push the token and the current state to the stack.
3. Go to the next state.

Example:




| | |
|---|---|
| Grammar | $S \to a\, b$ |
| Input | • a b $ |
| Stack | (empty) |
| Current State | 0 |

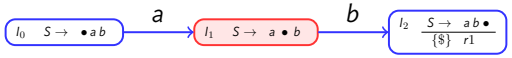
We will shift the a and then we go to state 1.

## Shifting

Shifting involves three steps.

1. Consume a token from the input.
2. Push the token and the current state to the stack.
3. Go to the next state.

Example:




Grammar $S \to a\, b$
Input a • b $
Stack 0, a
Current State 1

We will shift the b and then we go to state 2.

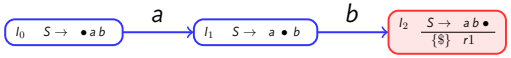


## Shifting

Shifting involves three steps.

1. Consume a token from the input.
2. Push the token and the current state to the stack.
3. Go to the next state.

Example:




Grammar $S \to a\, b$
Input a b • $
Stack 0, a, 1, b
Current State 2

What should happen now?

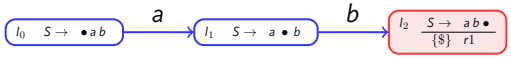


## Reducing

Reducing involves three steps.

1. Pop the tokens and states from the stack. (How many?)
2. Return to the last state popped.
3. Construct a new tree from the popped tokens.

Example:




Grammar $S \to a\, b$
Input a b • $
Stack 0, a, 1, b
Current State 2

We are ready to reduce.

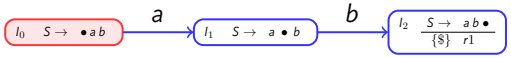


## Reducing

Reducing involves three steps.

1. Pop the tokens and states from the stack. (How many?)
2. Return to the last state popped.
3. Construct a new tree from the popped tokens.

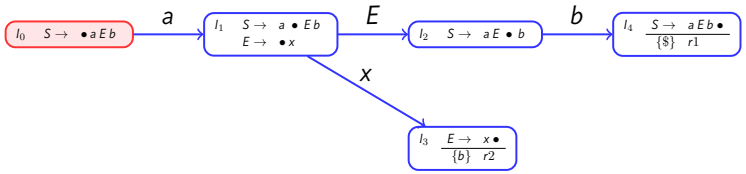
Example:




Grammar $S \to a\, b$
Input a b • $
Stack
Current State 0

Now we have an $S$ tree. Go To or Accept could happen here.

## A More Complex Example




Grammar

$S \to a\,E\,b$

$E \to x$

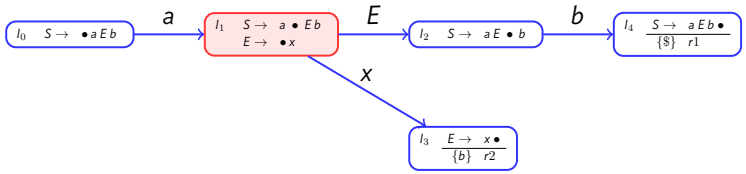
Input ● a x b $

Stack (Empty)

Current State 0



## A More Complex Example




Grammar

$S \to a\,E\,b$

$E \to x$

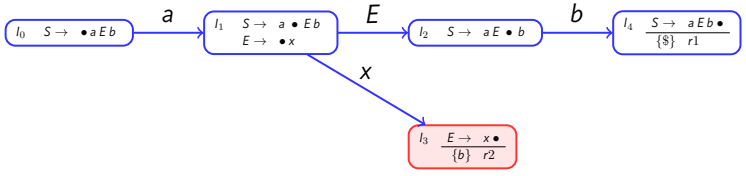
Input a ● x b $

Stack 0,a

Current State 1



## A More Complex Example




Grammar

$S \to a\,E\,b$

$E \to x$

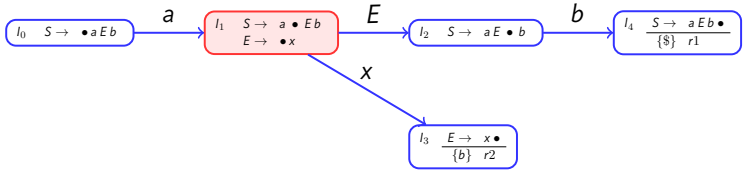
Input a x ● b $

Stack 0,a,1,x

Current State 4



## A More Complex Example




Grammar

$S \to a\,E\,b$

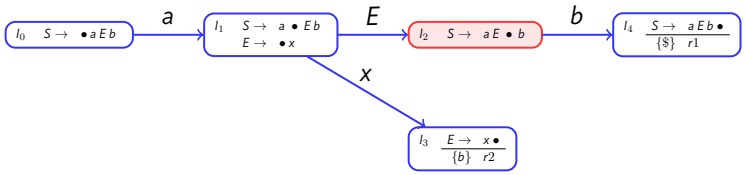
$E \to x$

Input a x ● b $

Stack 0,a

Current State 1

## A More Complex Example




Grammar

$S \to a\,E\,b$

$E \to x$

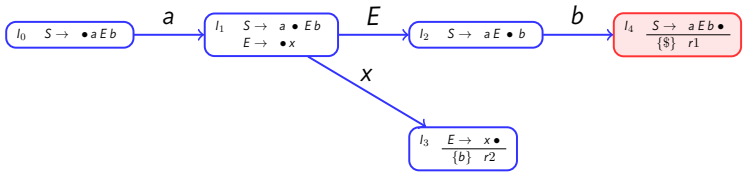
Input a x • b $

Current State 2

Stack 0,a, 1, x

Introduction
The Automata

## A More Complex Example




Grammar

$S \to a\,E\,b$

$E \to x$

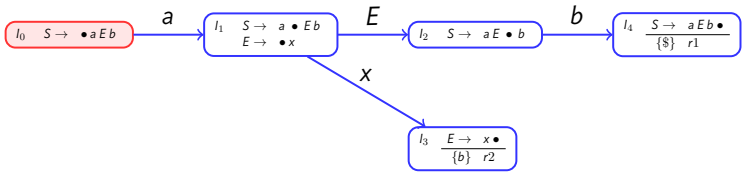
Input a x b • $

Current State 3

Stack 0,a, 1, x, 2, b

Introduction
The Automata

## A More Complex Example




Stack (Empty)

Grammar

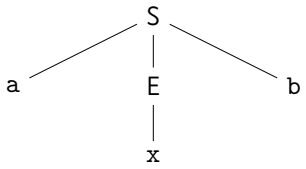
$S \to a\,E\,b$

$E \to x$

Input a x b • $

Current State 0

Now we have the result:




Introduction
The Automata

## Representing the Automata

We will represent the automata using two tables.

Action Table Shift, Reduce $n$, Accept

Goto Table Destination State

The rows are the state numbers, the columns are the symbols.

## The Algorithm

- To create the start state, add the *transitive closure* of the start symbol.

| Example 1 | | Start | | Example 2 | | Start | |
|---|---|---|---|---|---|---|---|
| $S \rightarrow$ | $x\ S\ e$ | $S \rightarrow$ | $\bullet\ x\ S\ e$ | $S \rightarrow$ | $x\ S\ e$ | | |
| $\mid$ | $E\ x$ | $\mid$ | $\bullet\ E\ x$ | $\mid$ | $F\ x$ | $S \rightarrow$ | $\bullet\ x\ S\ e$ |
| $E \rightarrow$ | $a\ E$ | $E \rightarrow$ | $\bullet\ a\ E$ | $E \rightarrow$ | $a\ E$ | $\mid$ | $\bullet\ F\ x$ |
| $\mid$ | $F\ x$ | $\mid$ | $\bullet\ F\ x$ | $\mid$ | $F\ x$ | $F \rightarrow$ | $\bullet\ q$ |
| $F \rightarrow$ | $q$ | $F \rightarrow$ | $\bullet\ q$ | $F \rightarrow$ | $q$ | | |



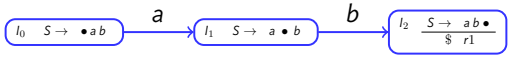
## The Algorithm, ctd

- Let $x$ be an arbitrary terminal, $A$ be an aribtrary nonterminal, and $\alpha$ and $\beta$ be arbitrary (possibly empty) strings of symbols.
- In an item set $i$, take every production of the form $E \rightarrow \alpha \bullet x\beta$ and produce a new state $j$ containing the transitive closure of $E \rightarrow \alpha x \bullet \beta$. Add a shift in the action table for column $x$ and state $i$, and destination state $j$ in the goto table for column $x$ and state $i$.
- In an item set $i$, take every production of the form $E \rightarrow \alpha \bullet A\beta$ and produce a new state $j$ containing the transitive closure of $E \rightarrow \alpha A \bullet \beta$. Add $j$ to the goto table in column $A$ and state $i$.
- In an item set $i$, take every rule of the form $E \rightarrow \alpha\bullet$ and add a reduce actions for state $i$ for each terminal in the follow set of $E$.
- If an item set is recreated, reuse the original; do not create a duplicate.



## Automata Example 1

### Automata




### Tabular Representation

Grammar

$S \rightarrow a\ b$

Action

| | a | b | $ |
|---|---|---|---|
| 0 | s | | |
| 1 | | s | |
| 2 | | | r1 |

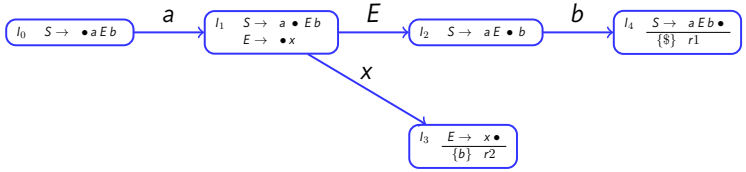
Goto

| | a | b | $ | S |
|---|---|---|---|---|
| 0 | 1 | | | |
| 1 | | 2 | | |
| 2 | | | | |



## Automata Example 2

### Automata




Grammar

$S \rightarrow a\ E\ b$
$E \rightarrow x$

Action

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | | s | |
| 2 | | s | | |
| 3 | | r2 | | |
| 4 | | | | r1 |

Goto

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | | 3 | | | 2 |
| 2 | | 4 | | | | |
| 3 | | | | | | |
| 4 | | | | | | |

## Automata Example 3.1

- Let's build the table for this automata.

$$
\begin{aligned}
S &\rightarrow a\,E\,b \\
&\mid a\,b\,S \\
E &\rightarrow E\,x \\
&\mid b
\end{aligned}
$$



## Automata Example 3.2

$I_0$ $S \rightarrow \bullet a\,E\,b$
$\bullet a\,b\,S$

Grammar

$$
\begin{aligned}
S &\rightarrow a\,E\,b \\
&\mid a\,b\,S \\
E &\rightarrow E\,x \\
&\mid b
\end{aligned}
$$

Action

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | | | | |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

Goto

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | | | | | | |
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.2

$I_0$ $S \rightarrow \bullet a\,E\,b \Leftarrow$
$\bullet a\,b\,S \Leftarrow$

Grammar

$$
\begin{aligned}
S &\rightarrow a\,E\,b \\
&\mid a\,b\,S \\
E &\rightarrow E\,x \\
&\mid b
\end{aligned}
$$

Action

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | | | | |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

Goto

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | | | | | | |
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.3

$I_0$ $S \rightarrow \bullet a\,E\,b$
$\bullet a\,b\,S$

a

$I_1$ $S \rightarrow a \bullet E\,b$
$a \bullet b\,S$
$E \rightarrow \bullet E\,x$
$\bullet b$

$$
\begin{aligned}
S &\rightarrow a\,E\,b \\
&\mid a\,b\,S \\
E &\rightarrow E\,x \\
&\mid b
\end{aligned}
$$

Action

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

Goto

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |

## Automata Example 3.3

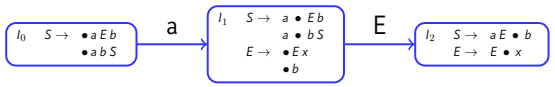

$$S \to a\,E\,b \mid a\,b\,S$$
$$E \to E\,x \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.4



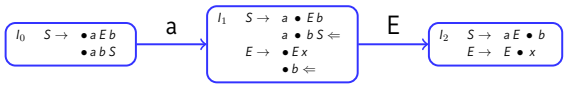

$$S \to a\,E\,b \mid a\,b\,S$$
$$E \to E\,x \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | | | | | 2 |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.4



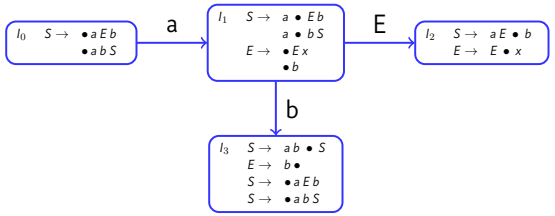

$$S \to a\,E\,b \mid a\,b\,S$$
$$E \to E\,x \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | | | | | 2 |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.5

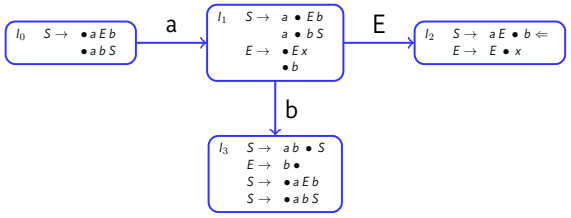



$$S \to a\,E\,b \mid a\,b\,S$$
$$E \to E\,x \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |

## Automata Example 3.5



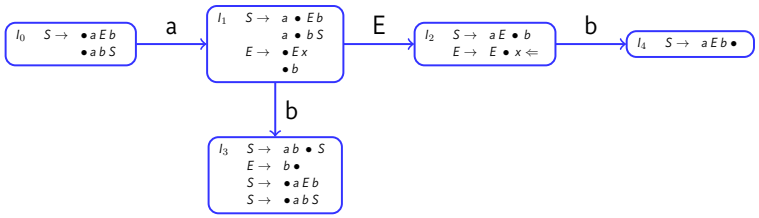

$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.6




$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.6



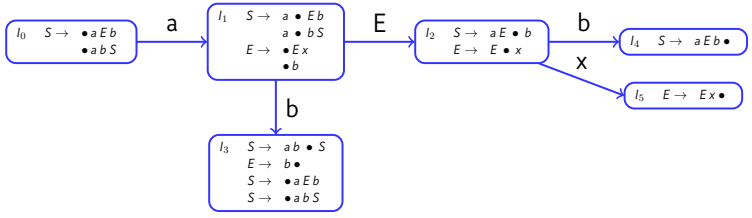

$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.7

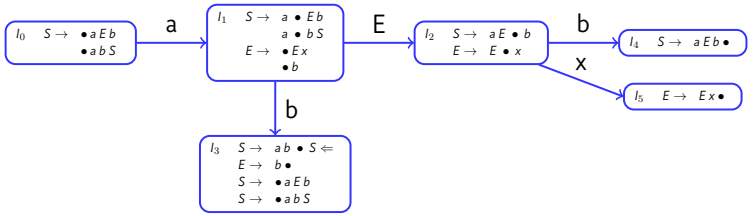



$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |

## Automata Example 3.7



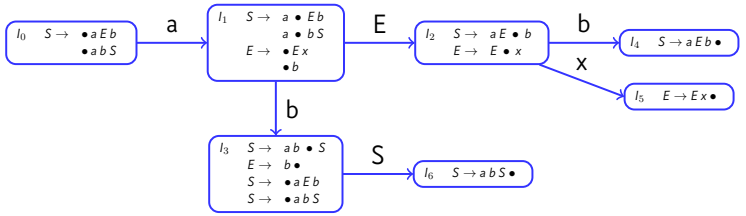

$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.8



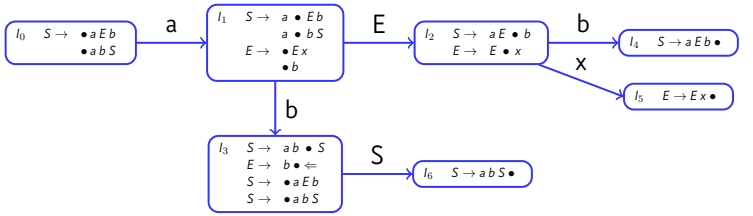

$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.8



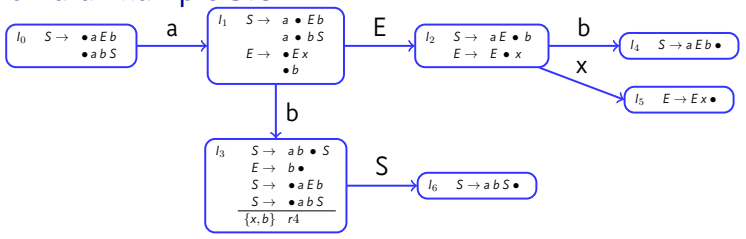

$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.8

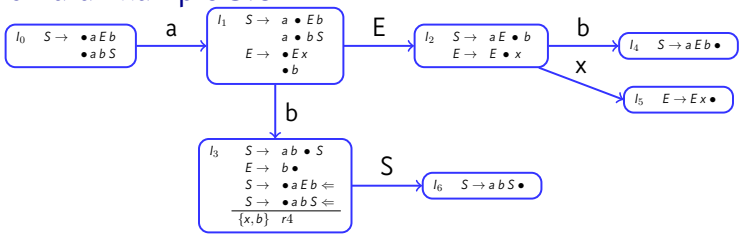



$$S \to aEb \mid abS$$
$$E \to Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | r4 | r4 | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |

## Automata Example 3.8



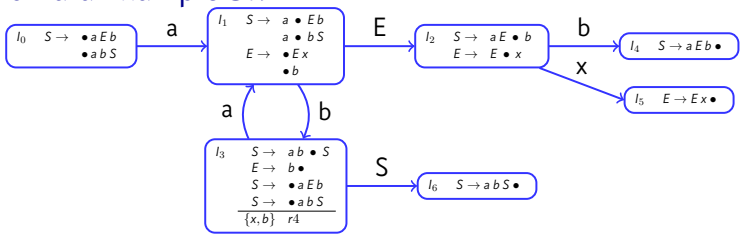

$$S \rightarrow aEb \mid abS$$
$$E \rightarrow Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | | r4 | r4 | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.9



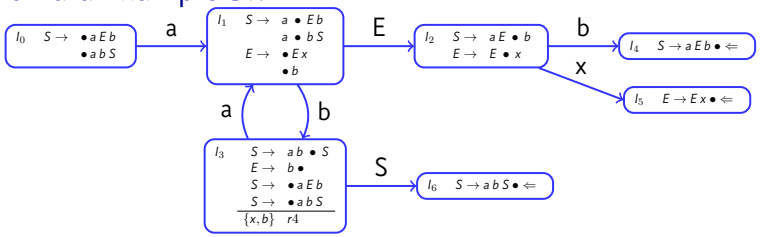

$$S \rightarrow aEb \mid abS$$
$$E \rightarrow Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | s | r4 | r4 | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | 1 | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.9



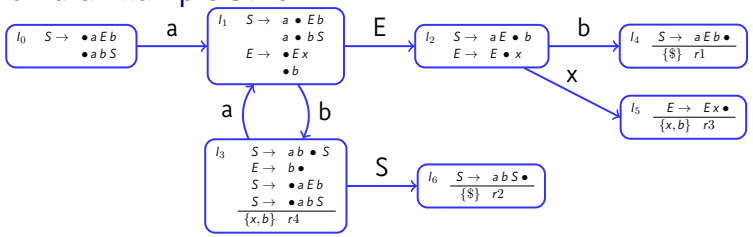

$$S \rightarrow aEb \mid abS$$
$$E \rightarrow Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | s | r4 | r4 | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | 1 | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |



## Automata Example 3.10




$$S \rightarrow aEb \mid abS$$
$$E \rightarrow Ex \mid b$$

| | a | b | x | $ |
|---|---|---|---|---|
| 0 | s | | | |
| 1 | | s | | |
| 2 | | s | s | |
| 3 | s | r4 | r4 | |
| 4 | | | | r1 |
| 5 | | r3 | r3 | |
| 6 | | | | r2 |

| | a | b | x | $ | S | E |
|---|---|---|---|---|---|---|
| 0 | 1 | | | | | |
| 1 | | 3 | | | | 2 |
| 2 | | 4 | 5 | | | |
| 3 | 1 | | | | 6 | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |

# Activity

Your turn. Try to build the automata for this grammar. There's a surprise waiting for you!

$$
\begin{array}{rcl}
S & \rightarrow & a\,E\,b \\
 & | & x \\
E & \rightarrow & E\,x\,E \\
 & | & b
\end{array}
$$