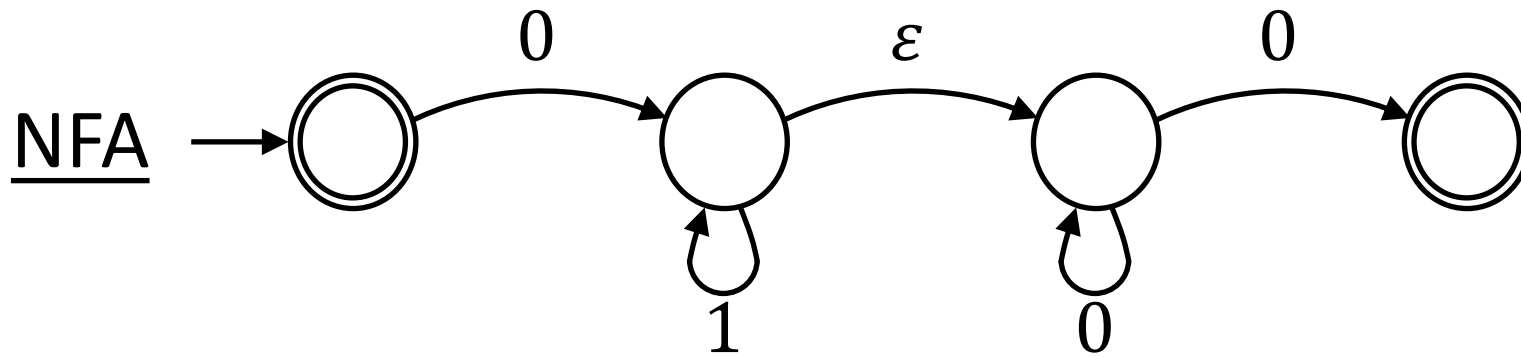


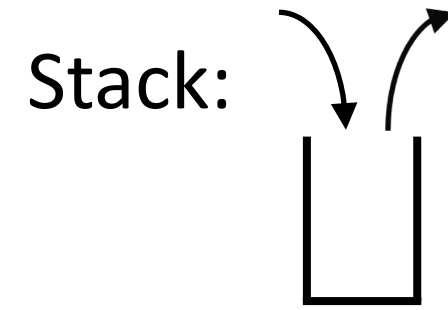
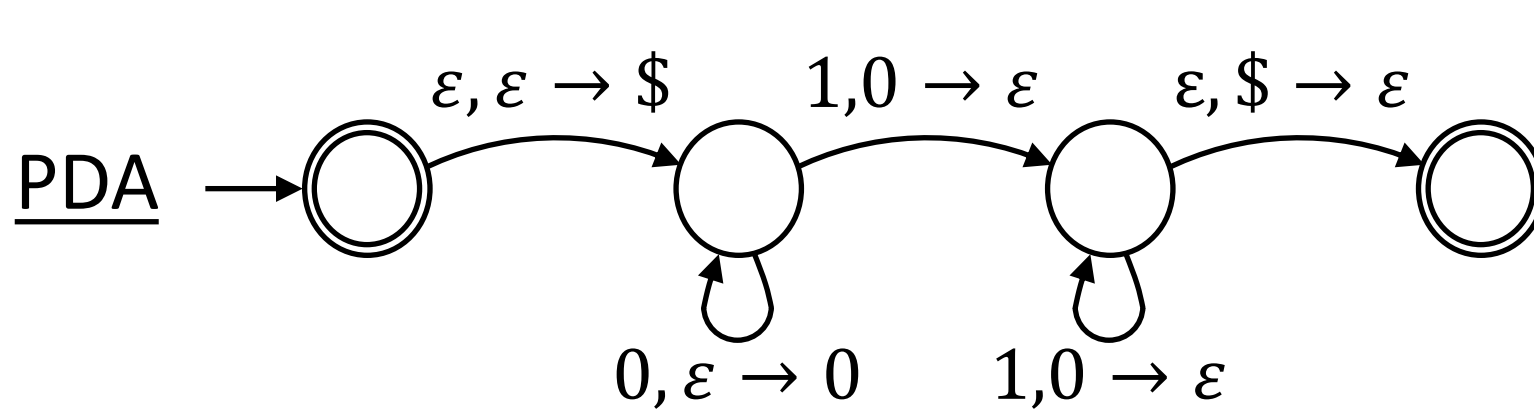
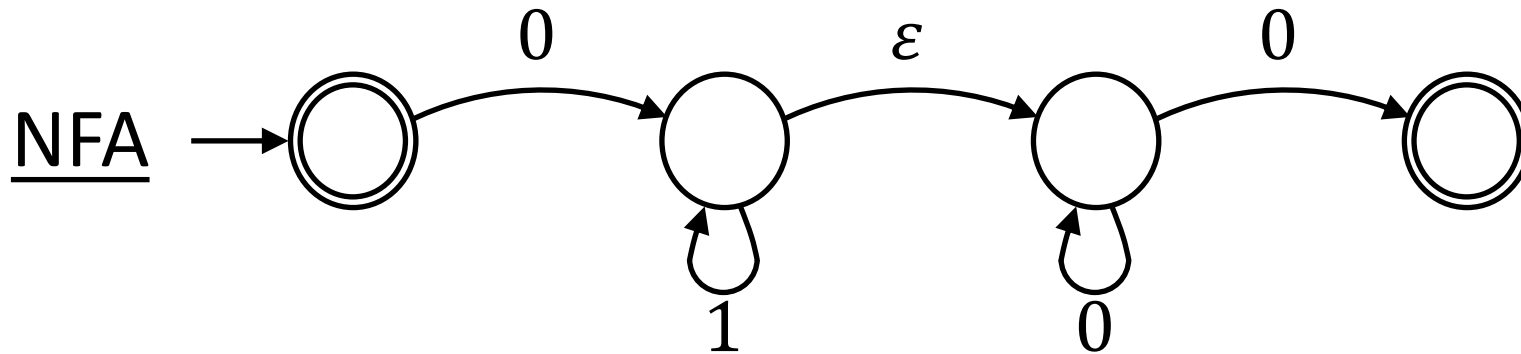
# Beyond Regular

## CSCI 338

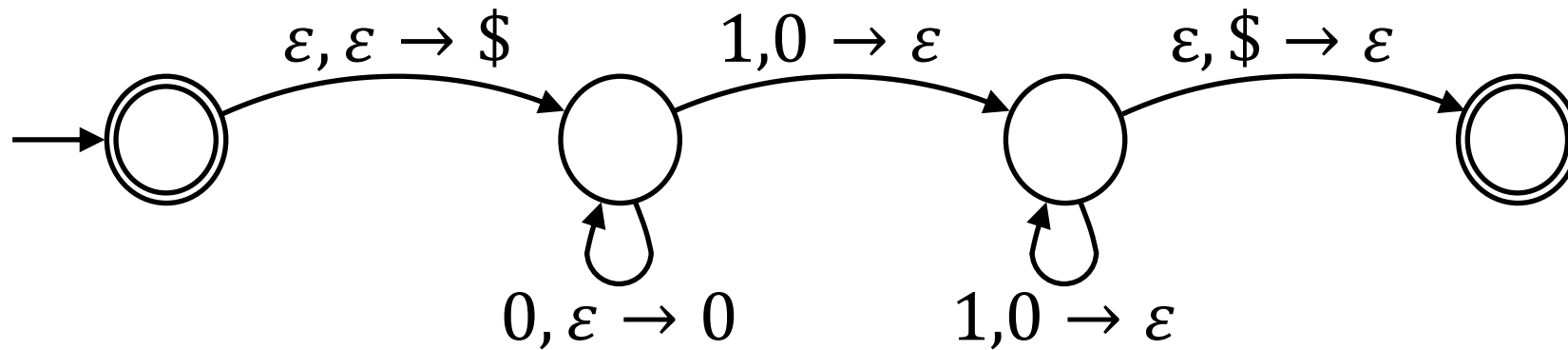
# Computational Models



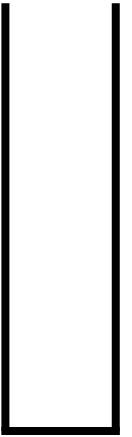
# Computational Models



# Pushdown Automata (PDA)



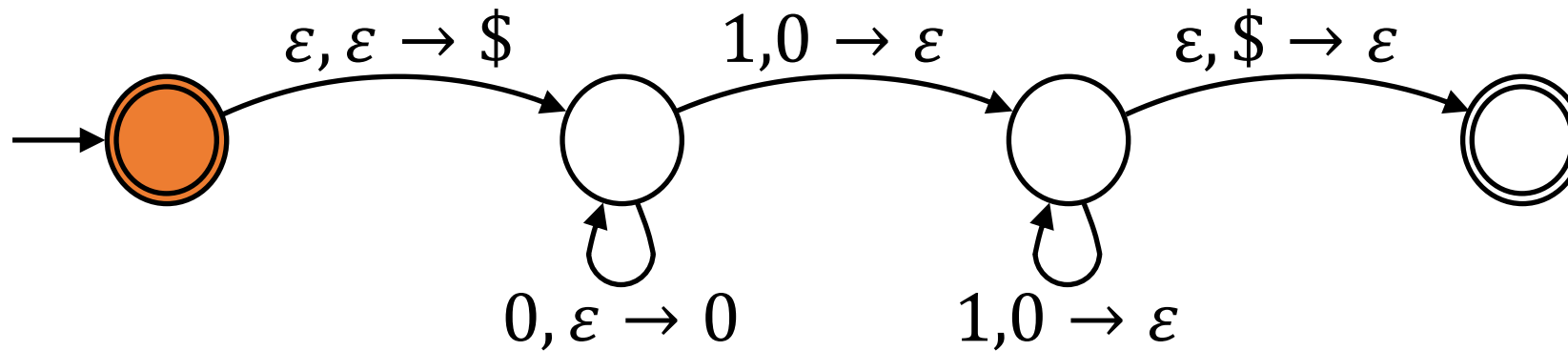
Stack:



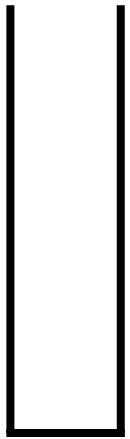
Transition Notation:

Input      Stack      Stack  
character , pop       $\rightarrow$       push

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

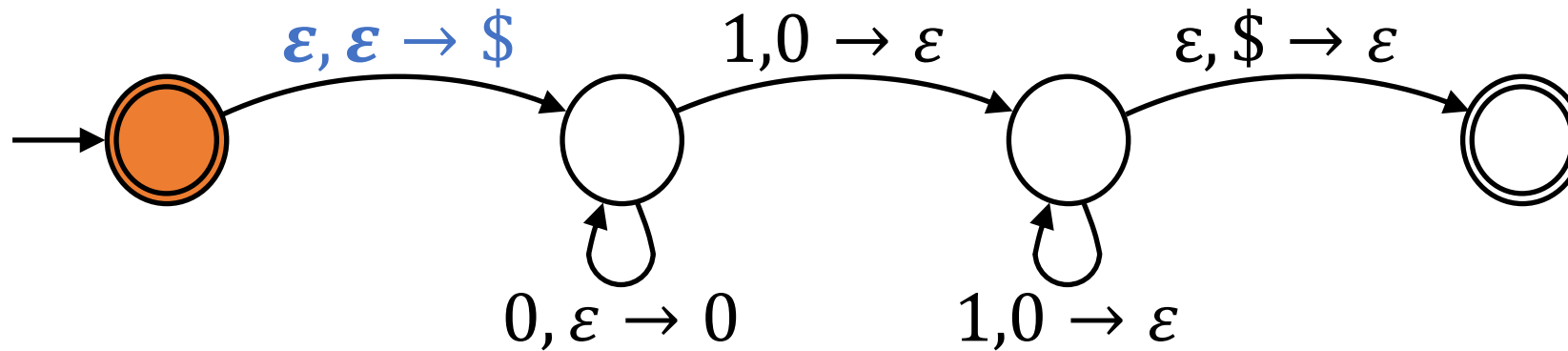
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

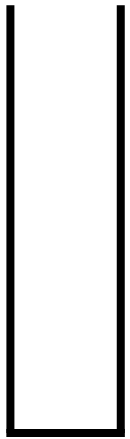
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

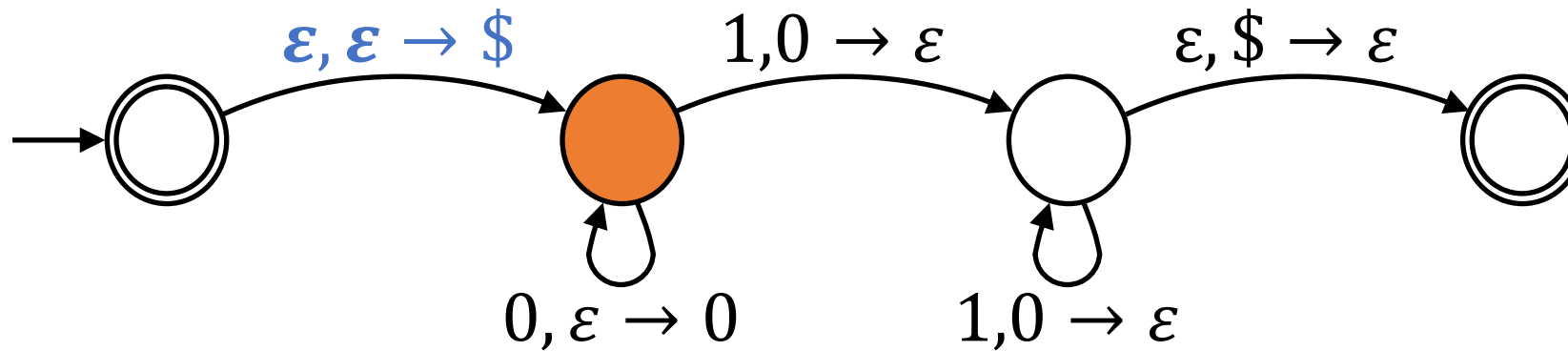
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

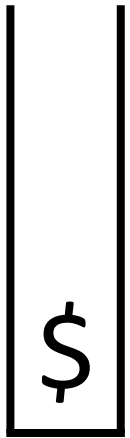
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

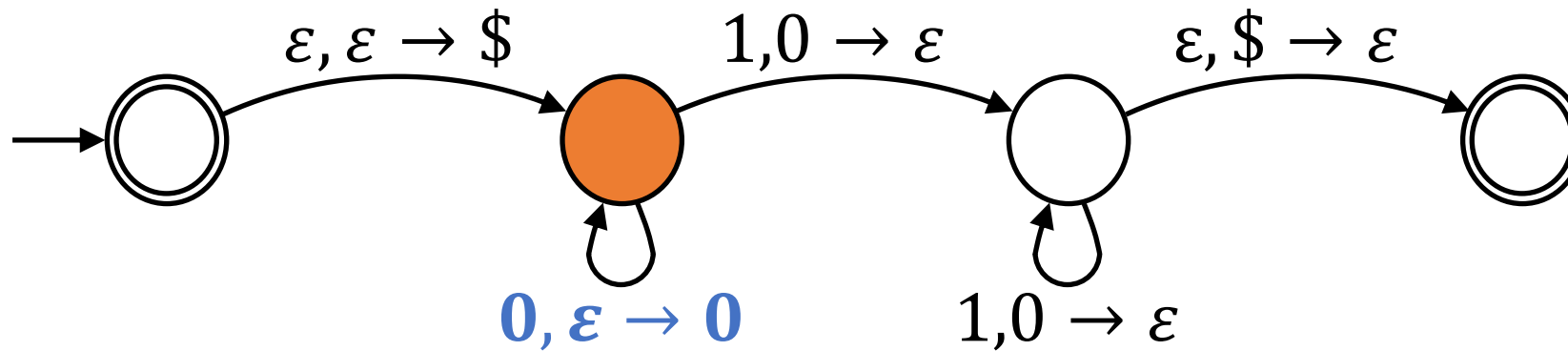
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

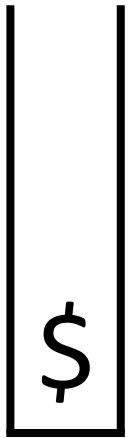
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

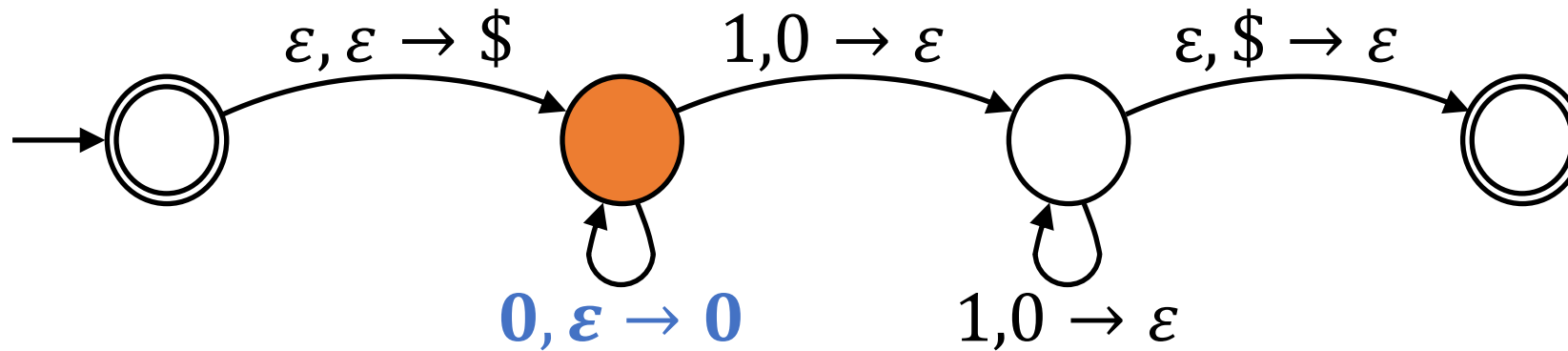
$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

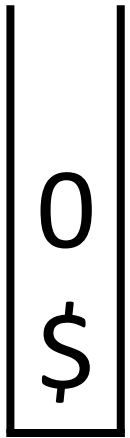
$s = 0011$



# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

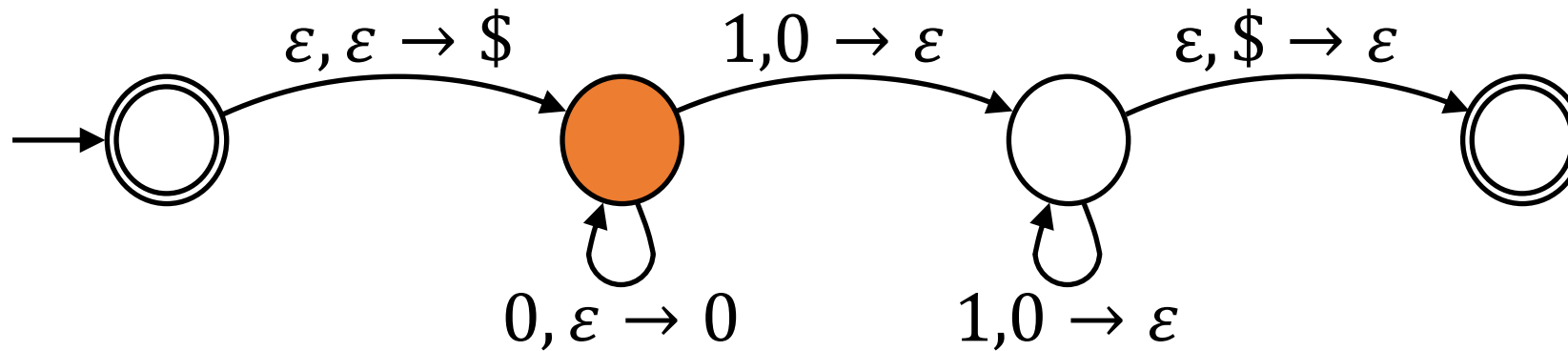
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

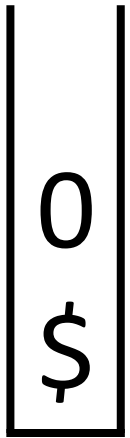
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

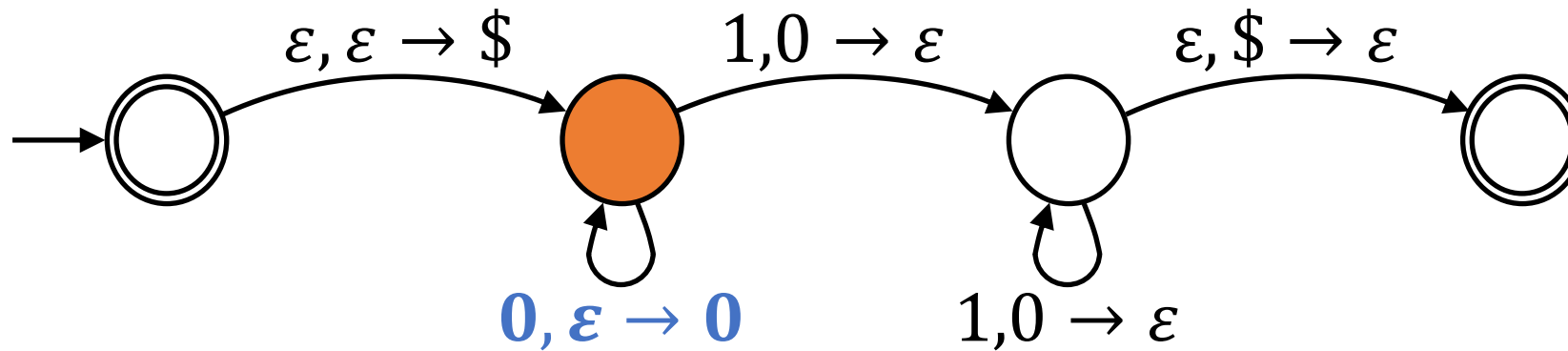
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 00\mathbf{0}11$

# Pushdown Automata (PDA)



Stack: 

0
0
\$

$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

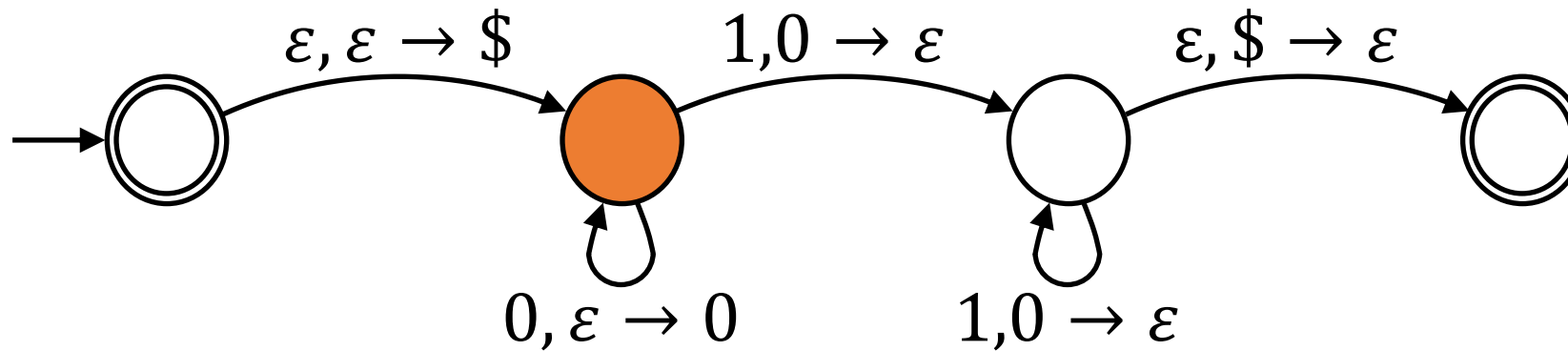
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 00\mathbf{0}11$

# Pushdown Automata (PDA)



Stack: 

0
0
\$

$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

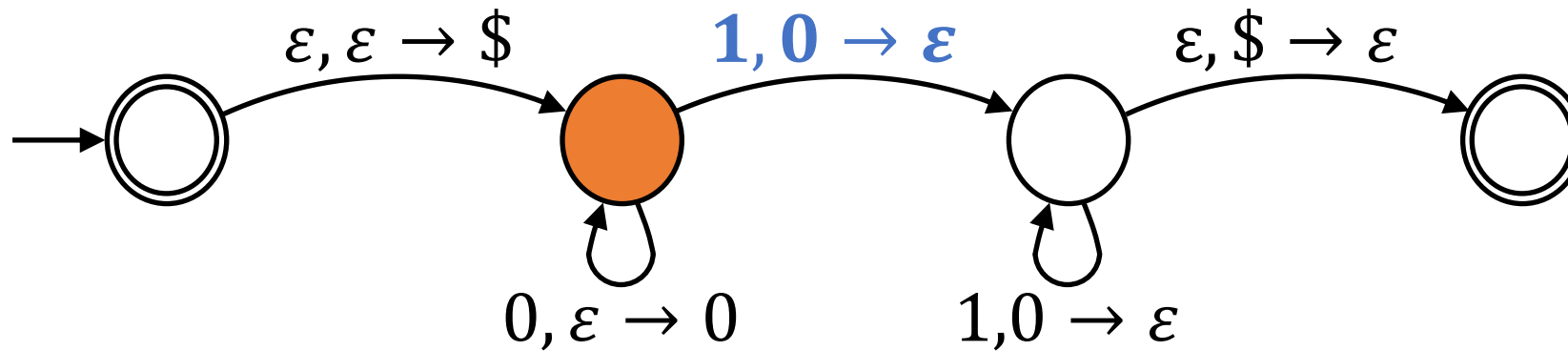
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 00\mathbf{1}1$

# Pushdown Automata (PDA)



Stack: 

0
0
\$

$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

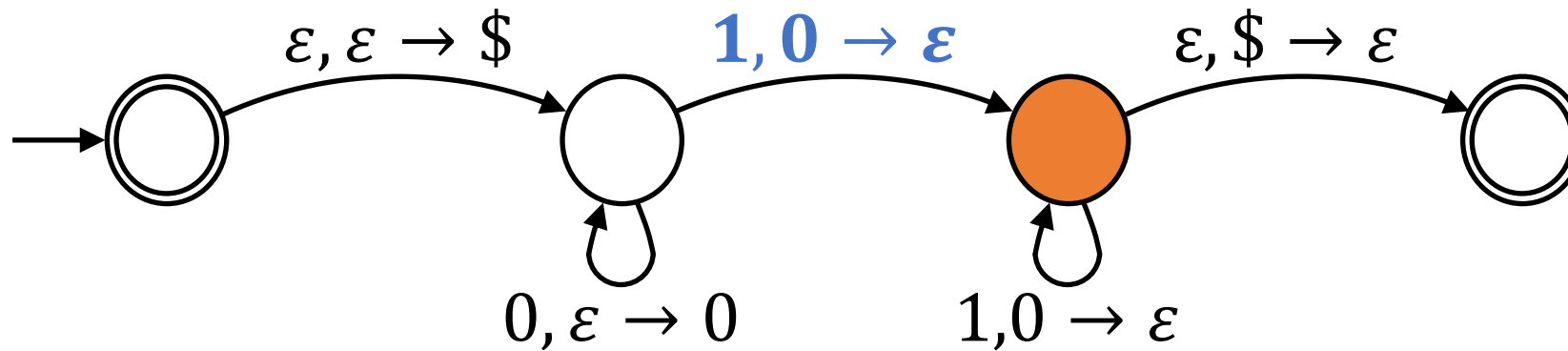
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

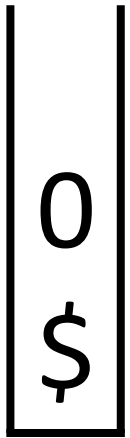
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 00\mathbf{1}1$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

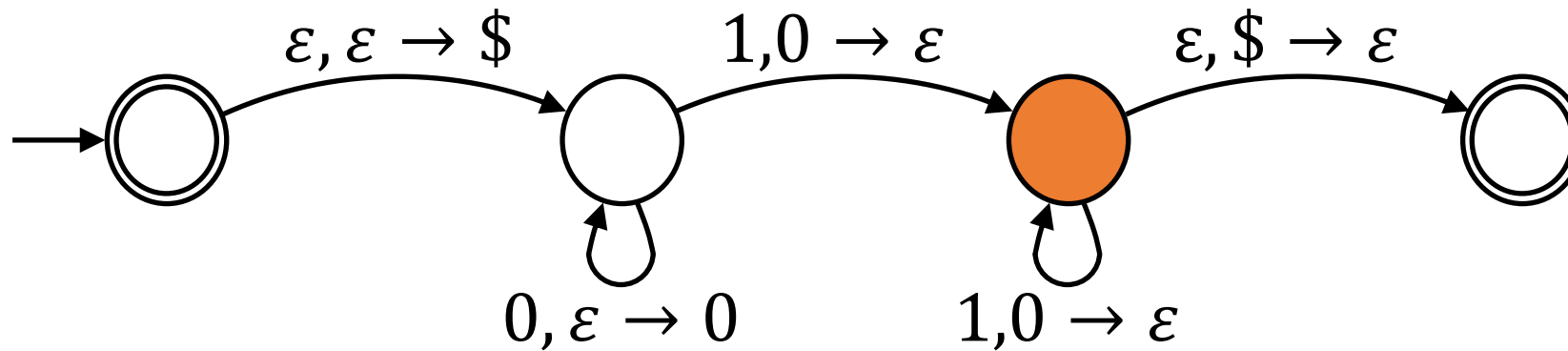
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

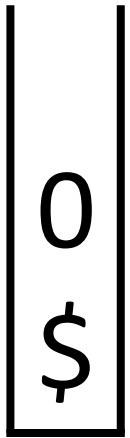
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

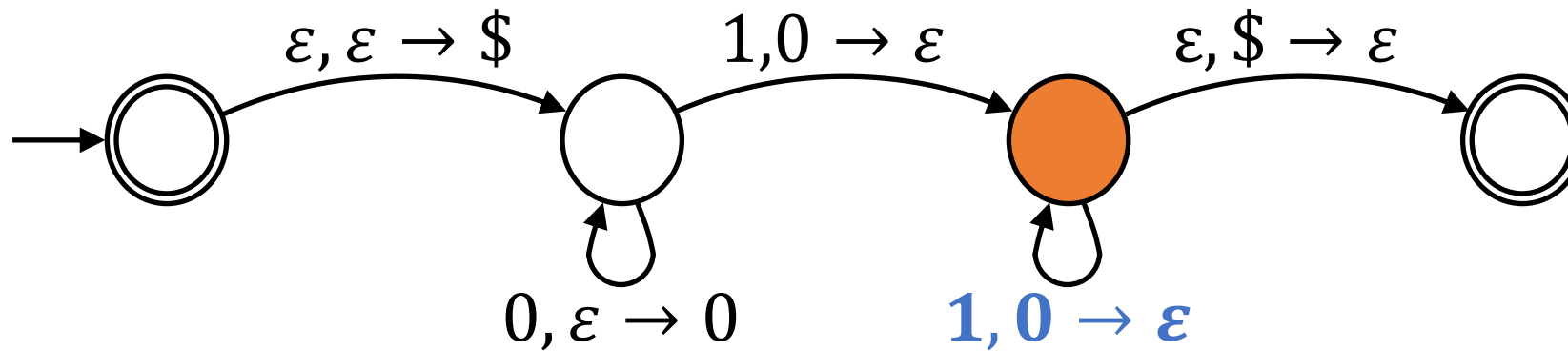
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

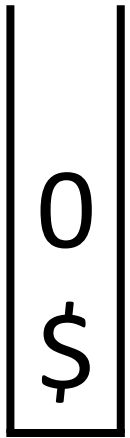
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

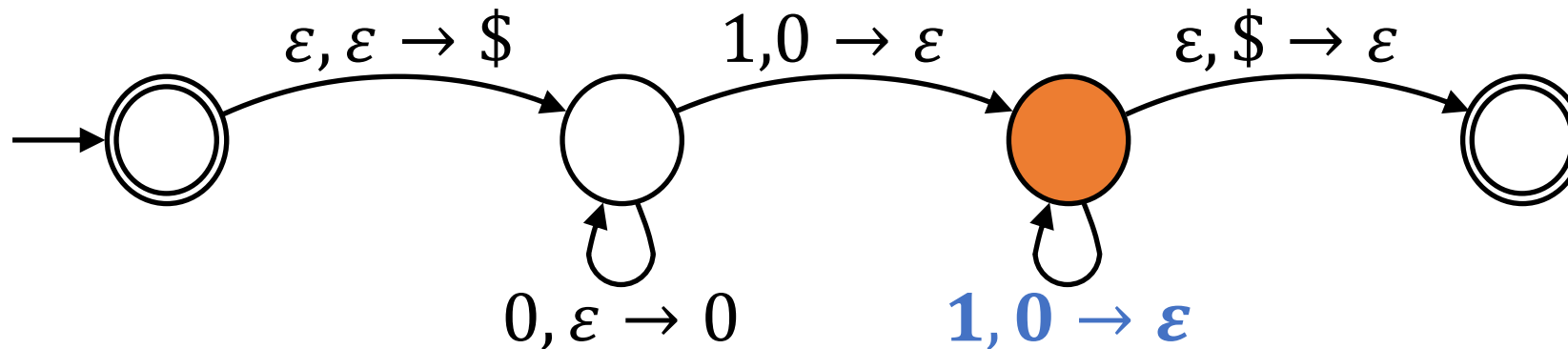
$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

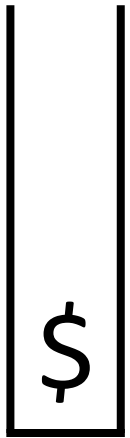
$s = 0011$



# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

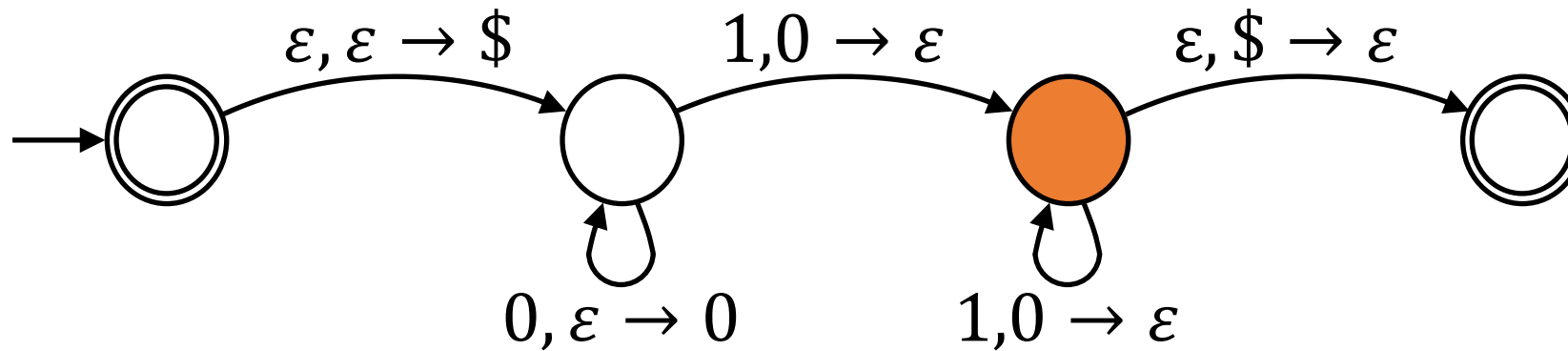
$\varepsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \varepsilon \rightarrow c$  If 'a' is read, push 'c'.

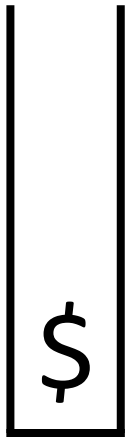
$a, b \rightarrow \varepsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

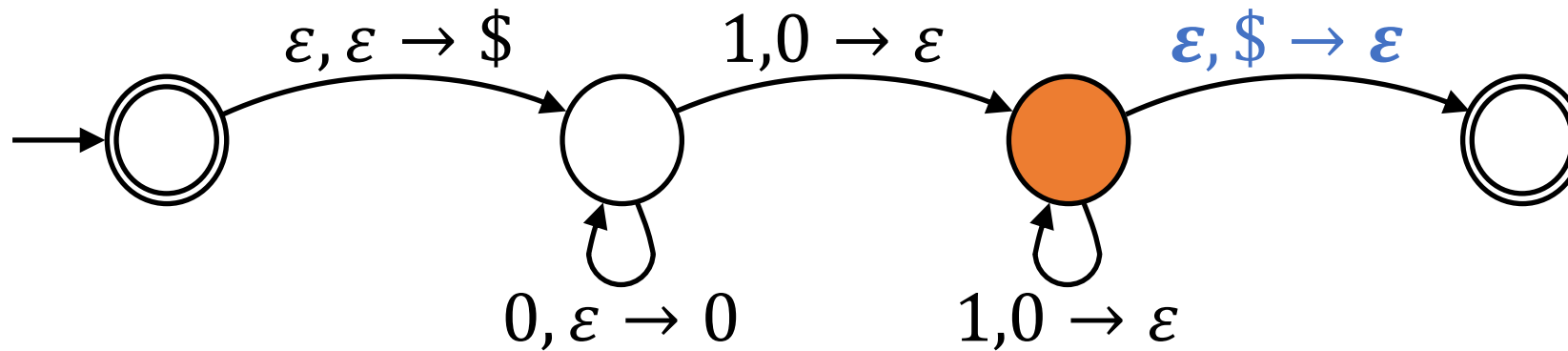
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

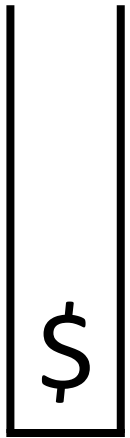
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

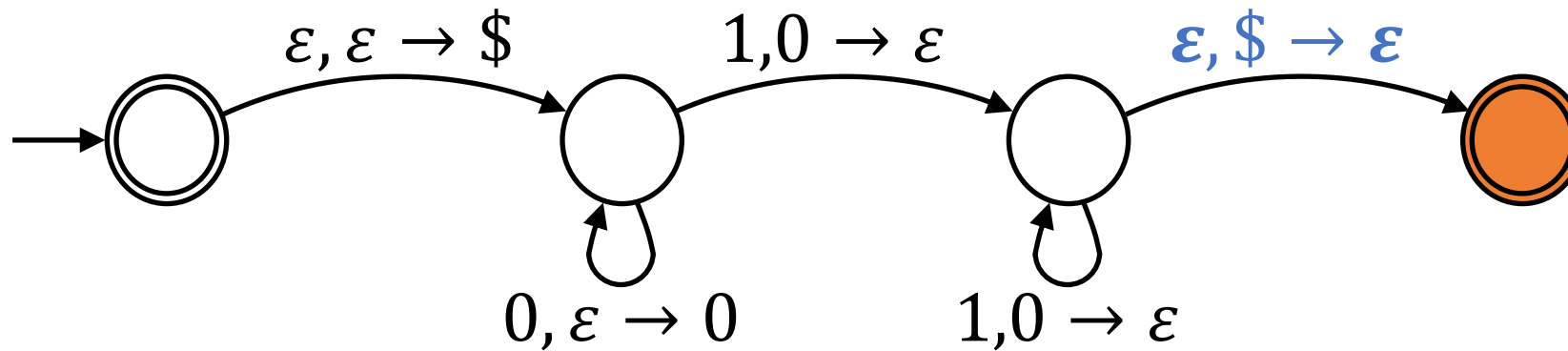
$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

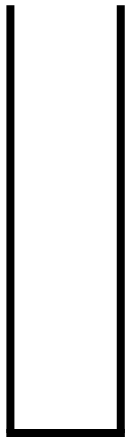
$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$

# Pushdown Automata (PDA)



Stack:



$a, b \rightarrow c$  If 'a' is read from input and 'b' is on top of stack, pop 'b' and push 'c'.

$\epsilon, b \rightarrow c$  Without reading input, if 'b' is on top of stack, pop 'b' and push 'c'.

$a, \epsilon \rightarrow c$  If 'a' is read, push 'c'.

$a, b \rightarrow \epsilon$  If 'a' is read and 'b' is on top of stack, pop 'b'.

$s = 0011$