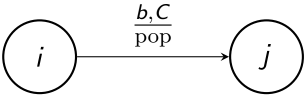
Pushdown Automata (Section 11.6)

Pushdown Automata

- A pushdown automaton (PDA) is a finite automaton with a stack that has stack operations pop, push, and nop. PDAs always start with one designated symbol on the stack. A state transition depends on the input symbol and the top of the stack. The machine then performs a stack operation and enters the next state.
- Representation can be graphical or with sets of 5-tuples. For example:



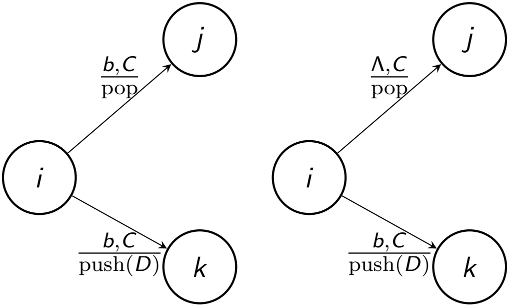
or (i, b, C, pop, j)

Execution of previous slide

- *Execution:* If the machine is in state i and the input letter is b and C is on the top of the stack, then pop the stack and enter state j.

Nondeterminism in PDAs

- *Nondeterminism* can occur in two ways as shown:

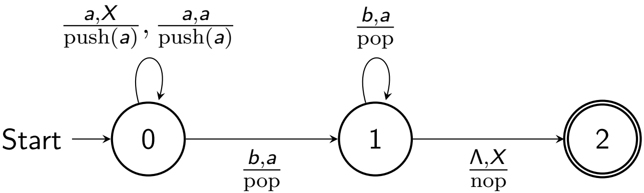


Acceptance

A string w is *accepted* by a PDA if there is a path from the start state to a final state such that the input symbols on the path edges concatenate to w. Otherwise, w is *rejected*.

Example

A PDA to accept the language $\{a^n b^n | n > 0\}$ as a graph and as a set of 5-tuples.



- $(0, a, X, \text{push}(a), 0)$
- $(0, a, a, \text{push}(a), 0)$
- $(0, b, a, \text{pop}, 1)$
- $(1, b, a, \text{pop}, 1)$
- $(1, \Lambda, X, \text{nop}, 2)$

Extension

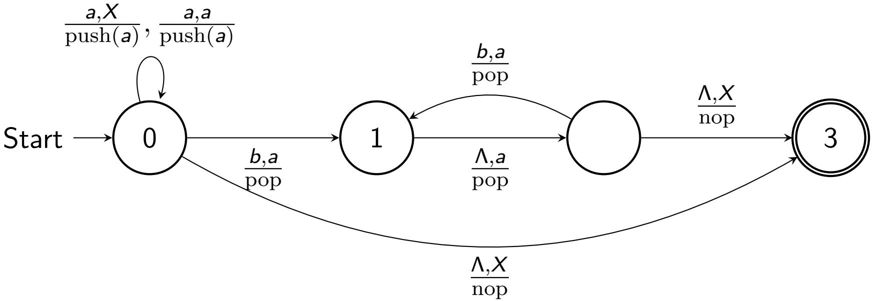
How would you modify the machine to accept $\{a^n b^n | n \in \mathbb{N}\}$?

Extension

How would you modify the machine to accept $\{a^n b^n | n \in \mathbb{N}\}$?
Answer: Add the instruction $(0, \Lambda, X, \mathrm{nop}, 2)$.

Another Example

Find a PDA to accept the language $\{a^{2n}b^n | n \in \mathbb{N}\}$.



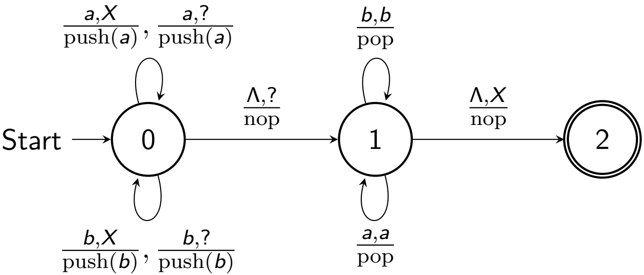
Context-Free Languages and PDAs

- *Theorem*: The context-free languages are exactly the languages accepted by PDAs.
- Proof could consist of showing how to transform a context-free grammar into a PDA, and showing how to transform a PDA into a context-free grammar. Both can be done, but we won't do them.

Nondeterminism Adds Power

- Nondeterministic PDAs are more powerful than deterministic PDAs.
- Consider the language of even palindromes over $\{a,b\}$. A context free grammar for this language is given by:
  - $S \rightarrow \Lambda | aSa | bSb$
- Any PDA to accept the language must make a nondeterministic decision to start comparing the second half of a string with the reverse of the first half.

Example



$\frac{a,X}{\text{push}(a)}, \frac{a,?}{\text{push}(a)}$
$\frac{b,b}{\text{pop}}$
$\frac{\Lambda,?}{\text{nop}}$
$\frac{\Lambda,X}{\text{nop}}$
Start
0
1
2
$\frac{b,X}{\text{push}(b)}, \frac{b,?}{\text{push}(b)}$
$\frac{a,a}{\text{pop}}$