A transition of the form $a, b \rightarrow z$ means “if the current input symbol is $a$ and the current stack symbol is $b$, then follow this transition, pop $b$, and push the string $z$”

Does this pushdown automaton accept the following strings?

- $[[[]]]$
- $[]$
A pushdown automaton is a six tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, where

- $Q$ is a finite set of states,
- $\Sigma$ is a finite alphabet (the input symbols),
- $\Gamma$ is a finite alphabet (the stack symbols),
- $\delta : (Q \times \Sigma_\epsilon \times \Gamma_\epsilon) \rightarrow \mathcal{P}(Q \times \Gamma_\epsilon)$ is the transition function,
- $q_0 \in Q$ is the initial state, and
- $F \subseteq Q$ is the set of accept states.
Balanced Brackets

Let $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, where

- $Q = \{q_1, q_2, q_3\}$,
- $\Sigma = \{[, ]\}$,
- $\Gamma = \{[, $ \}$,
- $q_0 = q_1$,
- $F = \{q_1, q_3\}$, and
- $\delta$ is given by the transition diagram:
Exercise

Build a pushdown automaton (PDA) to recognize the following languages

\[ A = \{ \, w w^R \mid w \in \{0,1\}^* \, \} \]

\[ B = \{ \, w \in \{0,1\}^* \mid w \text{ contains more 0's than 1's} \, \} \]
Finite Automaton and Pushdown Automaton
Theorem. A language is context-free if and only if it is recognized by some (non-deterministic) pushdown automaton.

Note. Unlike finite state machines, deterministic and nondeterministic PDA are not equivalent in power. Nondeterministic PDA are equivalent to CFGs in their power.
Are Regular Languages also CFL?
Non-Context-Free Languages

- Classic examples of languages that are not context-free

\[
L = \{a^n b^n c^n \mid n \geq 0\}
\]

\[
L = \{ww \mid w \in \{0, 1\}^*\}
\]

- Proved used generalization of pumping lemma
Closure Properties

- Union
- Complement
- Intersection
- Kleene Star