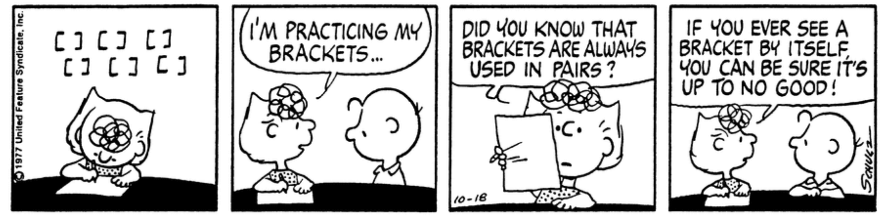
Pushdown Automata

Sipser: Section 2.2 pages 111 - 116

Hanging Out with the Wrong Crowd



Balanced Brackets

The grammar $G = (V, \Sigma, R, S)$, where

$V = \{S\}$,

$\Sigma = \{[,]\}$,

$R = \{ S \rightarrow \varepsilon \mid SS \mid [S] \}$

generates all strings of balanced brackets.

Is the language $L(G)$ regular? Why / Why not?

Recognizing Context-Free Languages

Grammars are ***language generators***. It is not immediately clear how they might be used as ***language recognizers***.

The language $L(G)$ of balanced brackets is not regular. It cannot be recognized by a finite state automaton.

However, it is very similar to the BEGIN/END blocks of many procedural languages and, therefore, must be recognized by some compiler or interpreter.

Auxiliary Store

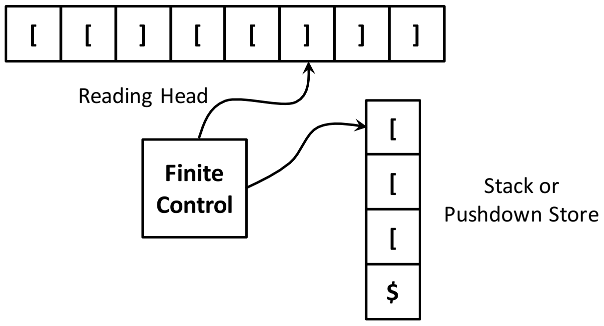
We could recognize the language $L(G)$ of balanced brackets by reading left to right, if we could remember left brackets along the way.

[[ ] [[ ]]]

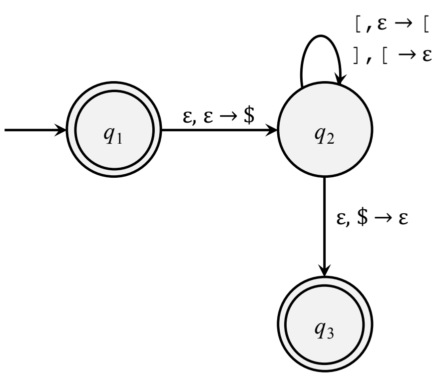
matches some previous
left bracket

Pushdown Automaton

The last left bracket seen matches the first right bracket.
This suggests a stack storage mechanism.



Describing a Pushdown Machine



Pushdown Automata

A ***pushdown automaton*** is a sextuple $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, where

- Q is a finite set of *states*,
- Σ is a finite alphabet (the *input symbols*),
- Γ is a finite alphabet (the *stack symbols*),
- $\delta: (Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon) \to P(Q \times \Gamma_\varepsilon)$ 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

$Q = \{q_1, q_2, q_3\}$,

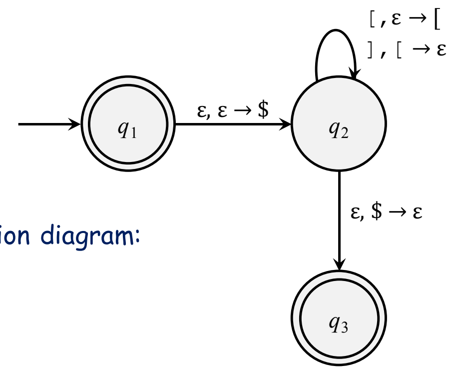
$\Sigma = \{[,]\}$,

$\Gamma = \{ [, \$ \}$,

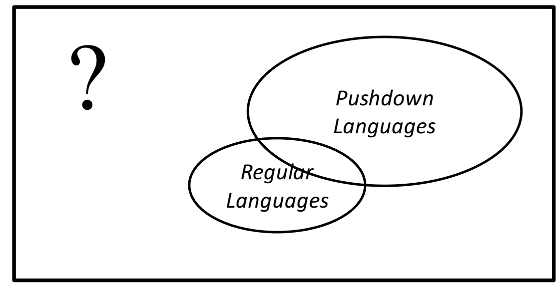
$q_0 = q_1$,

$F = \{q_1, q_3\}$, and

δ is given by the transition diagram:



Finite Automata and Pushdown Automata



Regular Languages ⇒ Pushdown Accept

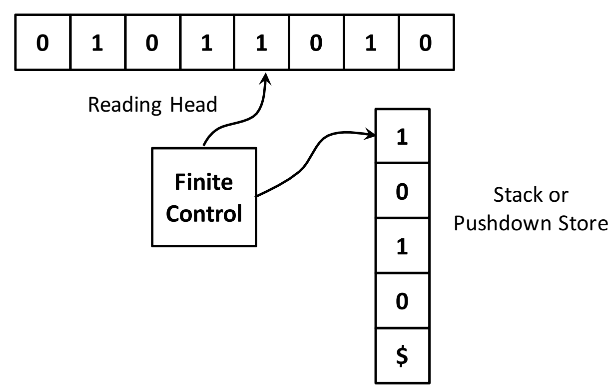
Proposition. Every finite automaton can be viewed as a pushdown automaton that never operates on its stack.

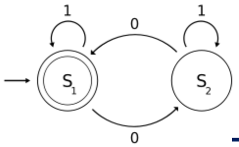
Proof. Let $M = (Q, \Sigma, \delta, q_0, F)$ be a finite automaton. Define $M' = (Q, \Sigma, \Gamma, \delta', q_0, F)$, where …

Pushdown Automata are Nondeterministic

Build a machine to recognize

$L(G) = \{ ww^R \mid w \in \{0,1\}^* \}$





Pushdown Automata are Nondeterministic

Build a machine to recognize

$L(G) = \{ a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k \}$