Lecture 12 — Pushdown Automata (1)

CSE205: Theory of Computation

Minseok Jeon

2026 Fall

Roadmap of This Course

We study three families of computational models of increasing power:

- **Finite automata (FA):** the basic model of computation.
- **Pushdown automata (PDA):** an extension of FA with a stack.
- **Turing machines (TM):** an extension of PDA.

This lecture introduces the pushdown automaton.

Pushdown Automata

Essentially, a PDA is an $\boldsymbol{\varepsilon}$-NFA equipped with a *stack*.

- In a finite automaton, the next state is determined by
  - the current state, and
  - the input symbol.
- In a pushdown automaton, the next state is determined by
  - the current state,
  - the input symbol, and
  - the symbol on top of the stack.

The stack gives the PDA an unbounded memory, so it can recognize languages such as $\{\boldsymbol{a^n b^n} \mid \boldsymbol{n} \geq 0\}$ that no FA can.

Formal Definition of Pushdown Automata

Definition (Pushdown Automaton)

A pushdown automaton (PDA) is a 7-tuple

$$P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$$

- Q: a finite set of states.
- Σ: a finite set of input symbols.
- Γ: a finite set of stack symbols (the stack alphabet).
- $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to 2^{Q \times \Gamma^*}$: the transition function.
- $q_0 \in Q$: the initial state.
- $Z_0 \in \Gamma$: the start stack symbol. Initially the stack contains only this symbol.
- $F \subseteq Q$: the set of final (accepting) states.

The Transition Function

$$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to 2^{Q \times \Gamma^*}$$

δ takes a triple (q, a, X):

- q: the current state,
- a: the current input symbol (or ε),
- X: the symbol currently on top of the stack.

The output of δ is a *finite set* of pairs (p, γ):

- p: the next state,
- $\gamma \in \Gamma^*$: the string of stack symbols that replaces X on top of the stack.

We read an edge label $a, X/\gamma$ as: "on input a with X on top, move and replace X by γ."

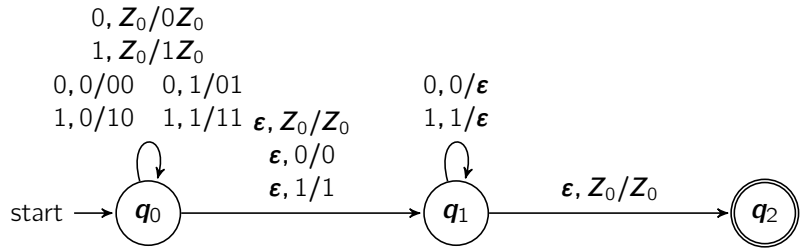
Example: A PDA for Even-Length Palindromes

$$P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_2\})$$

$$\begin{array}{rclrcl}
\delta(q_0, 0, Z_0) & = & \{(q_0, 0Z_0)\} & \delta(q_0, 1, Z_0) & = & \{(q_0, 1Z_0)\} \\
\delta(q_0, 0, 0) & = & \{(q_0, 00)\} & \delta(q_0, 0, 1) & = & \{(q_0, 01)\} \\
\delta(q_0, 1, 0) & = & \{(q_0, 10)\} & \delta(q_0, 1, 1) & = & \{(q_0, 11)\} \\
\delta(q_0, \varepsilon, Z_0) & = & \{(q_1, Z_0)\} & \delta(q_0, \varepsilon, 0) & = & \{(q_1, 0)\} \\
\delta(q_0, \varepsilon, 1) & = & \{(q_1, 1)\} & \delta(q_1, 0, 0) & = & \{(q_1, \varepsilon)\} \\
\delta(q_1, 1, 1) & = & \{(q_1, \varepsilon)\} & \delta(q_1, \varepsilon, Z_0) & = & \{(q_2, Z_0)\}
\end{array}$$

In q_0 the PDA pushes the first half onto the stack; it guesses the middle (via an ε-move to q_1), then in q_1 pops while matching the second half; reaching Z_0 moves to the accepting state q_2.

Transition Graph



Edges are labeled $a, X/\gamma$: input symbol, top of stack, and the string that replaces the top.

Exercises: Design PDAs

Construct a PDA for each of the following languages.

1. $L = \{a^n b^n \mid n \geq 0\}$

2. $L = \{w \in \{a, b\}^* \mid n_a(w) = n_b(w)\}$
   (equal number of a's and b's)

3. $L = \{a^i b^j c^k \mid i, j, k \geq 0 \ \wedge \ (i = j \ \vee \ i = k)\}$

We will work through these in the next lecture.