

# Lecture 15 – Examples of Pushdown Automata

## COSE215: Theory of Computation

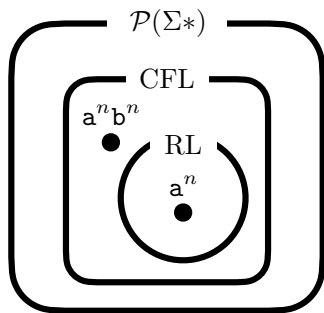
Jihyeok Park



2024 Spring

A **pushdown automaton (PDA)** is a finite automaton with a **stack**.

- Acceptance by **final states**
- Acceptance by **empty stacks**



Languages	Automata	Grammars
Context-Free Language (CFL)	Pushdown Automata (PDA)	Context-Free Grammar (CFG)
Regular Language (RL)	Finite Automata (FA)	Regular Expression (RE)

## 1. Examples of Pushdown Automata

Example 1:  $a^n b^n$

Example 2:  $a^n b^{2n}$

Example 3:  $ww^R$

Example 4: Balanced Parentheses

Example 5: Equal Number of a's and b's

Example 6: Unequal Number of a's and b's

Example 7: Not of the Form  $ww$

## Example 1: $a^n b^n$

Construct a PDA that accepts the language by **final states**:

$$L_F(P) = \{a^n b^n \mid n \geq 0\}$$

The key idea is to **count** the number of a's using the stack.

- 1 Start with the stack only having the initial stack alphabet  $Z$ .
- 2 Repeatedly **push**  $X$  onto the stack for each  $a$ .
- 3 Repeatedly **pop**  $X$  from the stack for each  $b$ .
- 4 Accept when the top of the stack is  $Z$ .

# Example 1: $a^n b^n$

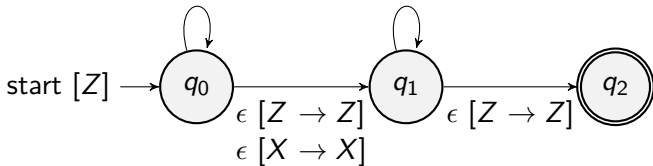
Construct a PDA that accepts the language by **final states**:

$$L_F(P) = \{a^n b^n \mid n \geq 0\}$$

$a [Z \rightarrow XZ]$

$a [X \rightarrow XX]$

$b [X \rightarrow \epsilon]$



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-an-bn-final.pdf>

## Example 2: $a^n b^{2n}$

Construct a PDA that accepts the language by **final states**:

$$L_F(P) = \{a^n b^{2n} \mid n \geq 0\}$$

Now, we need to push **two**  $X$ 's for each  $a$ .

- 1 Start with the stack only having the initial stack alphabet  $Z$ .
- 2 Repeatedly **push two**  $X$ 's onto the stack for each  $a$ .
- 3 Repeatedly **pop**  $X$  from the stack for each  $b$ .
- 4 Accept when the top of the stack is  $Z$ .

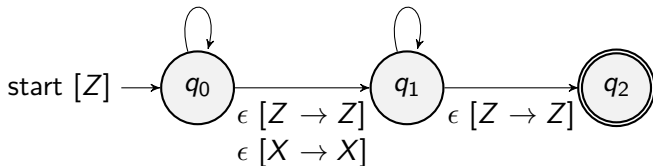
## Example 2: $a^n b^{2n}$

Construct a PDA that accepts the language by **final states**:

$$L_F(P) = \{a^n b^{2n} \mid n \geq 0\}$$

$a [Z \rightarrow XXZ]$

$a [X \rightarrow XXX] \quad b [X \rightarrow \epsilon]$



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-an-b2n-final.pdf>

## Example 3: $ww^R$

Construct a PDA that accepts the language by **final states**:

$$L_F(P) = \{ww^R \mid w \in \{a, b\}^*\}$$

The key idea is to **store** the first half of the word and **compare** it with the second half in reverse order using the stack.

- 1 Start with the stack only having the initial stack alphabet  $Z$ .
- 2 Repeatedly **push**  $X$  (or  $Y$ ) onto the stack for each  $a$  (or  $b$ ).
- 3 Repeatedly **pop**  $X$  (or  $Y$ ) from the stack for each  $a$  (or  $b$ ).
- 4 Accept when the top of the stack is  $Z$ .



# Example 3: $ww^R$

Construct a PDA that accepts the language by **final states**:

$$L_F(P) = \{ww^R \mid w \in \{a, b\}^*\}$$

$a [Z \rightarrow XZ]$

$a [X \rightarrow XX]$

$a [Y \rightarrow XY]$

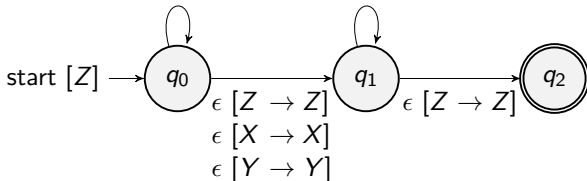
$b [Z \rightarrow YZ]$

$b [X \rightarrow YX]$

$b [Y \rightarrow YY]$

$a [X \rightarrow \epsilon]$

$b [Y \rightarrow \epsilon]$



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-w-wr-final.pdf>

## Example 4: Balanced Parentheses

Construct a PDA that accepts the language by **empty stacks**:

$$L_E(P) = \{w \in \{(, )\}^* \mid w \text{ is balanced}\}$$

The key idea is to **count** the number of **unmatched** open parentheses.

- 1 Start with the stack only having the initial stack alphabet  $Z$ .
- 2 If the current symbol is  $($ , push  $($  onto the stack.
- 3 If the current symbol is  $)$ , pop  $($  from the stack.
- 4 Repeat steps 2 and 3.
- 5 Accept when the top of the stack is  $Z$ .

## Example 4: Balanced Parentheses

Construct a PDA that accepts the language by **empty stacks**:

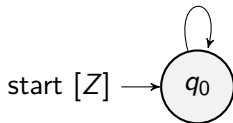
$$L_E(P) = \{w \in \{ (, ) \}^* \mid w \text{ is balanced}\}$$

( [Z → (Z]

( [( → ((]

) [( → ε]

ε [Z → ε]



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-balanced-empty.pdf>

## Example 5: Equal Number of a's and b's

Construct a PDA that accepts the language by **empty stacks**:

$$L_E(P) = \{w \in \{a, b\}^* \mid N_a(w) = N_b(w)\}$$

where  $N_a(w)$  and  $N_b(w)$  are the number of a's and b's in  $w$ , respectively.

Consider the following function  $f : \{a, b\}^* \rightarrow \mathbb{N}$ :

$$f(w) = N_a(w) - N_b(w)$$

The key idea is to represent the **positive value** of  $f(w)$  using the number of  $P$ 's and the **negative value** of  $f(w)$  using the number of  $N$ 's.

- 1 Start with the stack only having the initial stack alphabet  $Z$ .
- 2 If the current symbol is a, **push**  $P$  or **pop**  $N$ .
- 3 If the current symbol is b, **push**  $N$  or **pop**  $P$ .
- 4 Repeat steps 2 and 3.
- 5 Accept when the top of the stack is  $Z$ .

## Example 5: Equal Number of a's and b's

Construct a PDA that accepts the language by **empty stacks**:

$$L_E(P) = \{w \in \{a, b\}^* \mid N_a(w) = N_b(w)\}$$

where  $N_a(w)$  and  $N_b(w)$  are the number of a's and b's in  $w$ , respectively.

$a$   $[Z \rightarrow PZ]$

$a$   $[P \rightarrow PP]$

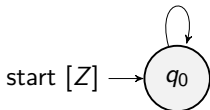
$a$   $[N \rightarrow \epsilon]$

$b$   $[Z \rightarrow NZ]$

$b$   $[P \rightarrow \epsilon]$

$b$   $[N \rightarrow NN]$

$\epsilon$   $[Z \rightarrow \epsilon]$



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-eq-a-b-empty.pdf>

## Example 6: Unequal Number of a's and b's

Construct a PDA that accepts the language by **empty stacks**:

$$L_E(P) = \{w \in \{a, b\}^* \mid N_a(w) \neq N_b(w)\}$$

where  $N_a(w)$  and  $N_b(w)$  are the number of a's and b's in  $w$ , respectively.

The key idea is same but we accept the top of the stack is  $P$  or  $N$ .

- 1 Start with the stack only having the initial stack alphabet  $Z$ .
- 2 If the current symbol is a, **push**  $P$  or **pop**  $N$ .
- 3 If the current symbol is b, **push**  $N$  or **pop**  $P$ .
- 4 Repeat steps 2 and 3.
- 5 Accept when the top of the stack is  $P$  or  $N$ .

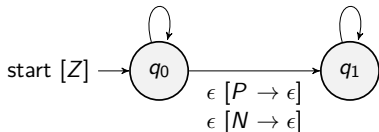
## Example 6: Unequal Number of a's and b's

Construct a PDA that accepts the language by **empty stacks**:

$$L_E(P) = \{w \in \{a, b\}^* \mid N_a(w) \neq N_b(w)\}$$

where  $N_a(w)$  and  $N_b(w)$  are the number of a's and b's in  $w$ , respectively.

$a$	$[Z \rightarrow PZ]$	
$a$	$[P \rightarrow PP]$	
$a$	$[N \rightarrow \epsilon]$	
$b$	$[Z \rightarrow NZ]$	$\epsilon [Z \rightarrow \epsilon]$
$b$	$[P \rightarrow \epsilon]$	$\epsilon [P \rightarrow \epsilon]$
$b$	$[N \rightarrow NN]$	$\epsilon [N \rightarrow \epsilon]$



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-uneq-a-b-empty.pdf>

## Example 7: Not of the Form $ww$

Construct a PDA that accepts the language by **empty stacks**:

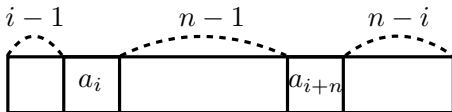
$$L_E(P) = \{x \in \{a, b\}^* \mid x \text{ is not of the form } ww\}$$

There are two cases of  $x \in L_E(P)$ :

- ①  $x$  is an **odd-length** word or
- ②  $x$  is divided into two **same-length** but **unequal** words.

In the second case, assume  $x = a_1 \cdots a_{2n}$ . Then,

$$\exists 1 \leq i \leq n. a_i \neq a_{i+n}$$





## Example 7: Not of the Form $ww$

Construct a PDA that accepts the language by **empty stacks**:

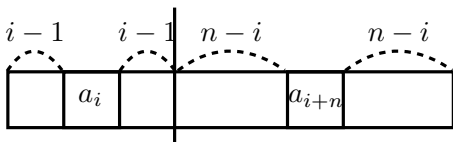
$$L_E(P) = \{x \in \{a, b\}^* \mid x \text{ is not of the form } ww\}$$

There are two cases of  $x \in L_E(P)$ :

- ①  $x$  is an **odd-length** word or
- ②  $x$  is divided into two **same-length** but **unequal** words.

In the second case, assume  $x = a_1 \cdots a_{2n}$ . Then,

$$\exists 1 \leq i \leq n. a_i \neq a_{i+n}$$



## Example 7: Not of the Form $ww$

Construct a PDA that accepts the language by **empty stacks**:

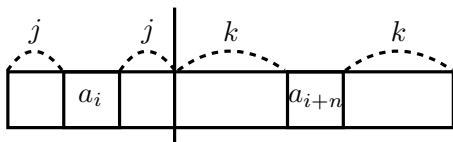
$$L_E(P) = \{x \in \{a, b\}^* \mid x \text{ is not of the form } ww\}$$

There are two cases of  $x \in L_E(P)$ :

- ①  $x$  is an **odd-length** word or
- ②  $x$  is divided into two **same-length** but **unequal** words.

In the second case, assume  $x = a_1 \cdots a_{2n}$ . Then,

$$\exists 1 \leq i \leq n. a_i \neq a_{i+n}$$



## Example 7: Not of the Form $ww$

Construct a PDA that accepts the language by **empty stacks**:

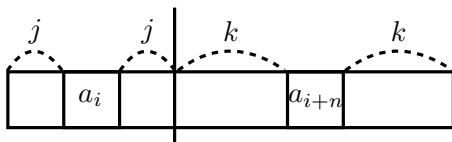
$$L_E(P) = \{x \in \{a, b\}^* \mid x \text{ is not of the form } ww\}$$

There are two cases of  $x \in L_E(P)$ :

- ①  $x$  is an **odd-length** word or
- ②  $x$  is divided into two **odd-length** words whose **centers** are different.

In the second case, assume  $x = a_1 \cdots a_{2n}$ . Then,

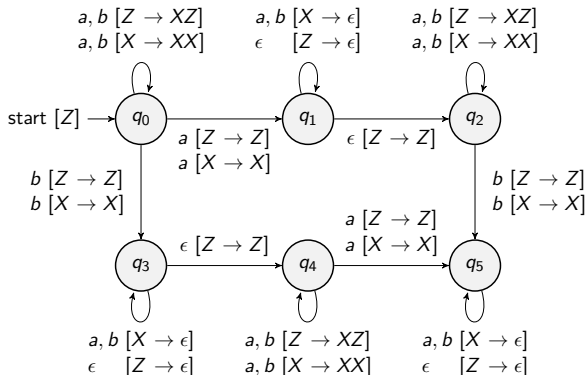
$$\exists 1 \leq i \leq n. a_i \neq a_{i+n}$$



## Example 7: Not of the Form $ww$

Construct a PDA that accepts the language by **empty stacks**:

$$L_E(P) = \{x \in \{a, b\}^* \mid x \text{ is not of the form } ww\}$$



<https://plrg.korea.ac.kr/courses/cose215/materials/pda-not-w-w-empty.pdf>

## 1. Examples of Pushdown Automata

Example 1:  $a^n b^n$

Example 2:  $a^n b^{2n}$

Example 3:  $ww^R$

Example 4: Balanced Parentheses

Example 5: Equal Number of a's and b's

Example 6: Unequal Number of a's and b's

Example 7: Not of the Form  $ww$

- Please see this document on GitHub:

<https://github.com/ku-plrg-classroom/docs/tree/main/cose215/pda-examples>

- The due date is 23:59 on May 15 (Wed.).
- Please only submit `Implementation.scala` file to [Blackboard](#).

- Equivalence of Pushdown Automata and Context-Free Grammars

Jihyeok Park

[jihyeok\\_park@korea.ac.kr](mailto:jihyeok_park@korea.ac.kr)

<https://plrg.korea.ac.kr>