Recall that PDA is non-deterministic.

We can actually define deterministic PDA (DPDA).

In Chapter 1, DFA $\equiv$ NFA

Both generate regular languages.

But $\text{PDA} \neq \text{DPDA}$

and therefore $\text{CFL} \neq \text{DCFL}$
DPDA was not discussed in earlier versions of the textbook
As this topic is less important, we will only explain what DPDA is without getting into more details
It’s more complicated to define DPDA than PDA
The reason is that in DPDA we must ensure the deterministic moves
Formal definition of DPDA I

- $(Q, \Sigma, \Gamma, \delta, q_0, F)$
  - $Q, \Sigma, \Gamma, F$: finite sets
  - $Q$: states
  - $\Sigma$: alphabet
  - $\Gamma$: stack alphabet
  - $\delta$:
    $$Q \times \Sigma_\epsilon \times \Gamma_\epsilon \rightarrow (Q \times \Gamma_\epsilon) \cup \{\emptyset\}$$
  - $q_0 \in Q$: start state
  - $F \subset Q$: set of accept states
Note for PDA

\[ \delta : Q \times \Sigma_{\epsilon} \times \Gamma_{\epsilon} \rightarrow P(Q \times \Gamma_{\epsilon}) \]

Also \( \delta \) satisfies \( \forall q \in Q, a \in \Sigma, x \in \Gamma, \) exactly one of

\[ \delta(q, a, x), \quad \delta(q, a, \epsilon), \quad \delta(q, \epsilon, x), \quad \delta(q, \epsilon, \epsilon) \]

is not \( \emptyset \)

Reason: at \( q \) all four can be taken at PDA

Rule: follow the one which is not \( \emptyset \)
Acceptance and rejection of DPDA I

- **Acceptance**: same as DFA.
  Reach an accept state after the last symbol
  Otherwise: reject

- **Rejection**: occurs if
  1. not at an accept state after the last symbol
     (same as DFA)
  2. DPDA fails to read the input
     1. pop an empty stack
     2. endless $\epsilon$-input moves

- **Example**: pop an empty stack
Acceptance and rejection of DPDA II

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>$\epsilon$</td>
<td>0</td>
</tr>
</tbody>
</table>

$q$ | $\emptyset$ | $\emptyset$ | $(q, \epsilon)$ | $\emptyset$

input $0, \emptyset$ is rejected: the only possible move is to pop up zero, but the stack is empty

- Example: fails to read the whole string

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>$\epsilon$</td>
<td>0</td>
</tr>
</tbody>
</table>

$q$ | $\emptyset$ | $\emptyset$ | $\emptyset$ | $(q, 1)$

input $0, \emptyset$ is rejected: endless $\epsilon$-input

$0, \emptyset \rightarrow 0, \{1\}, \rightarrow 0, \{1, 1\} \rightarrow \ldots$