Theory of Computation: CS-202

Outline

- Two Important Normal Forms
  - Chomsky Normal Form
  - Greibach Normal Form
- Push Down Automata
  - Deterministic Push Down Automata
  - Non Deterministic Push Down Automata

Chomsky Normal Form

- A Context free Grammar G is in Chomsky normal form if all the productions are of the form

$$A \rightarrow BC$$

or $A \rightarrow a$

Where, $A, B, C \in V$ and $a \in T$

Example

Consider the grammar G with production:

$S \rightarrow AS \mid a$

$A \rightarrow SA \mid b$

is in Chomsky Normal Form.

Example

Consider the grammar G with production:

$S \rightarrow AS \mid AAS$

$A \rightarrow SA \mid aS$

is not in Chomsky Normal Form.

Example

Convert the grammar G=({A,B,C}, {a,b,c}, S, P) into Chomsky Normal Form.

P: $S \rightarrow AB\ a$

$A \rightarrow aab$

$B \rightarrow Ac$

$\Rightarrow$ $S \rightarrow ABB_a$

$A \rightarrow B_a B_a B_b$

$B \rightarrow A B_c$

$B_a \rightarrow a, \quad B_b \rightarrow b, \quad B_c \rightarrow c$

$\Rightarrow$ $S \rightarrow AD_1$

$D_1 \rightarrow BB_a$

$A \rightarrow D_2 B_b$

$D_2 \rightarrow B_a B_a$, $B \rightarrow A B_c$

$B_a \rightarrow a, \quad B_b \rightarrow b, \quad B_c \rightarrow c$

Greibach Normal Form

A Context free Grammar G is in Greibach Normal Form if all the productions are of the form

$A \longrightarrow a\alpha$

or $A \rightarrow a$

Where, $A \in V$ and $a \in T$, $\alpha \in V^*$

Example

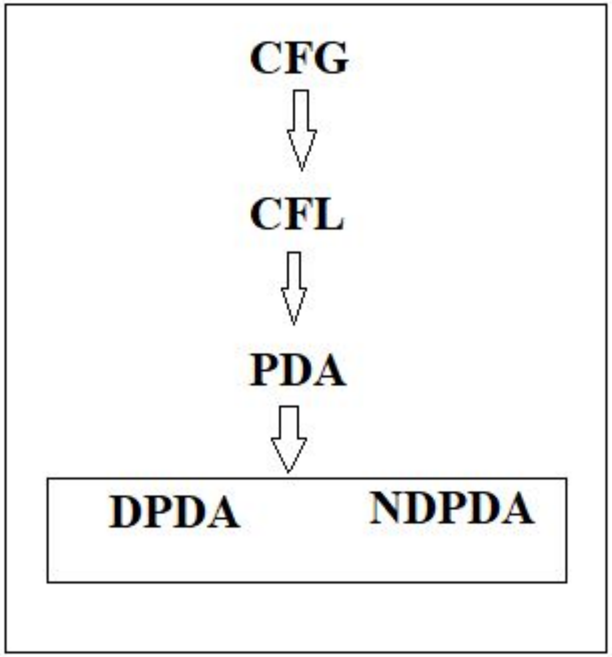
Construct a grammar in Greibach normal form equivalent to the grammar
$S \to AB,\ A \to \mathrm{aA} \mid bB|b, \quad B \to \mathrm{b}$

The given grammar is not in Greibach normal form However, using the substitution rule, we can immediately get the equivalent grammar

$$S \to aAB|bBB|bB$$
$$A \to aA|bB|b$$
$$B \to b$$

Which is in Greibach normal form

Context free Grammar, Language and PDA



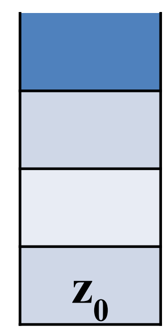
Formal Definition of a deterministic PDA

A pushdown automaton (PDA) is defined by the seven-tuples:

$M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$

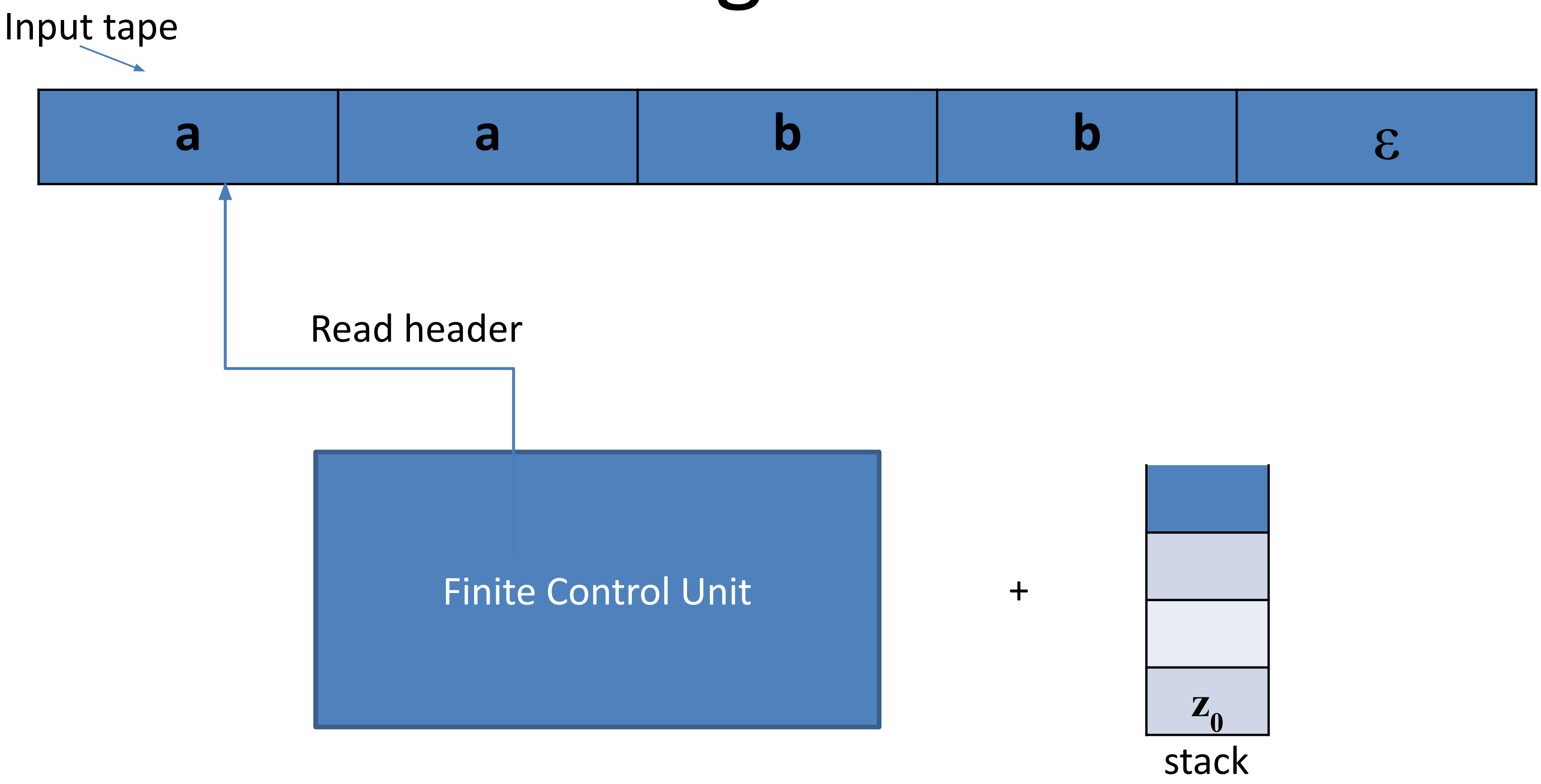
Q	A finite set of states
Σ	A finite set of input alphabet
Γ	A finite set of stack alphabet
q_0	The initial/starting state, q_0 is in Q
z_0	A starting stack symbol, is in Γ
F	A set of final/accepting states, which is a subset of Q
δ	A transition function, where

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



stack

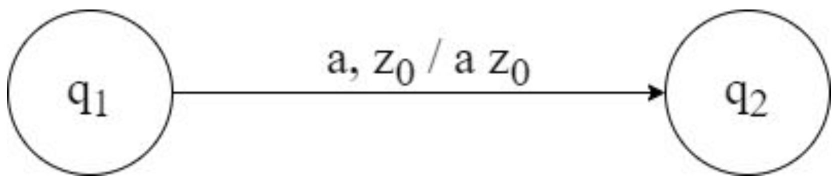
Block diagram of PDA

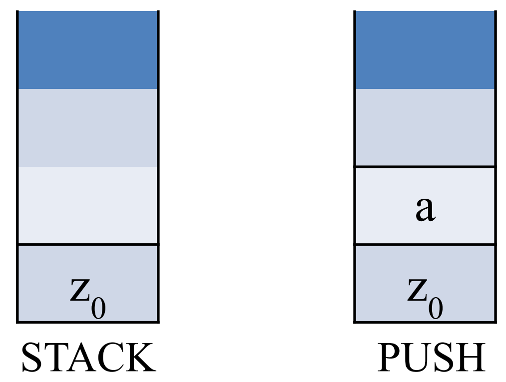


Moves of the PDA: Push, Pop, Skip

Push:

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



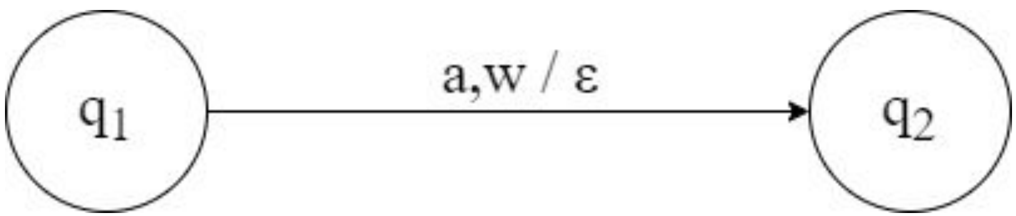


$\delta(q_1, a, z_0) = (q_2, az_0)$

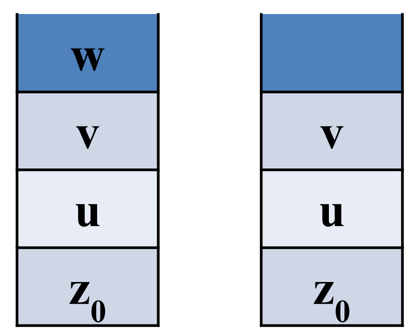
Moves of the PDA (Cont..)

Pop:

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

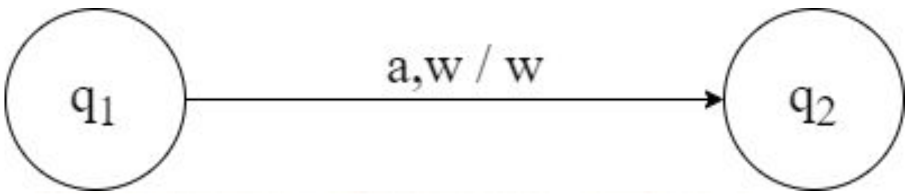


$\delta(q_1, a, w) = (q_2, \varepsilon)$



Moves of the PDA (Cont..)

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



$\delta(q_1, a, w) = (q_2, w)$

Moves of the PDA (Cont..)

Final:

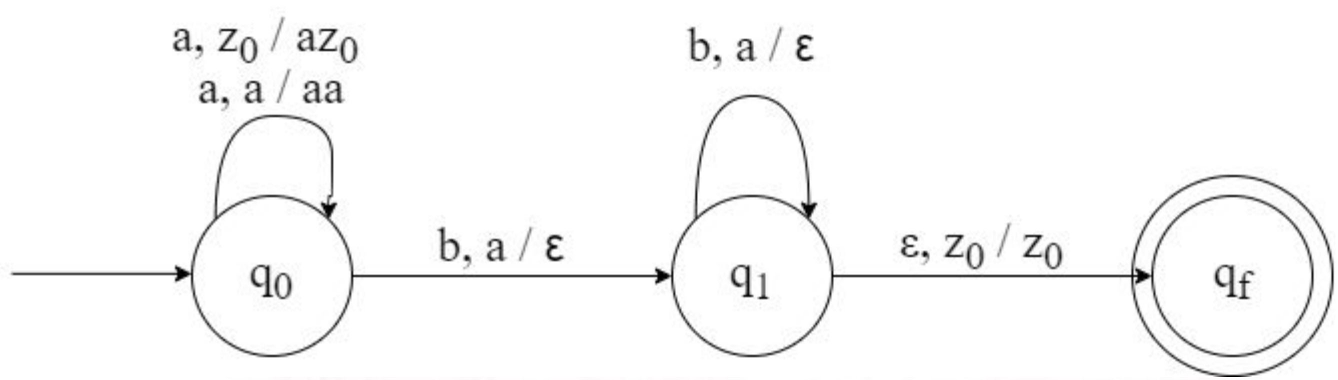
δ: Q x (Σ U {ε}) x $\Gamma \rightarrow$ Q x Γ^*

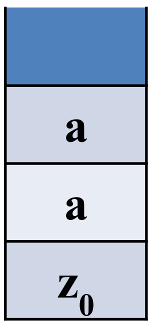
$\delta(q_1, \varepsilon, z_0) = (q_f, z_0)$

Or $\delta(q_1, \varepsilon, z_0) = (q_f, \varepsilon)$

Design a PDA for the language $L=\{a^n b^n, n \geq 1\}$

a	a	b	b	ε





Stack

Design a PDA for the language $L=\{a^n b^n, n \geq 1\}$

Transition steps for L

$\delta(q_0, a, z_0) = (q_0, az_0)$

$\delta(q_0, a, a) = (q_0, aa)$

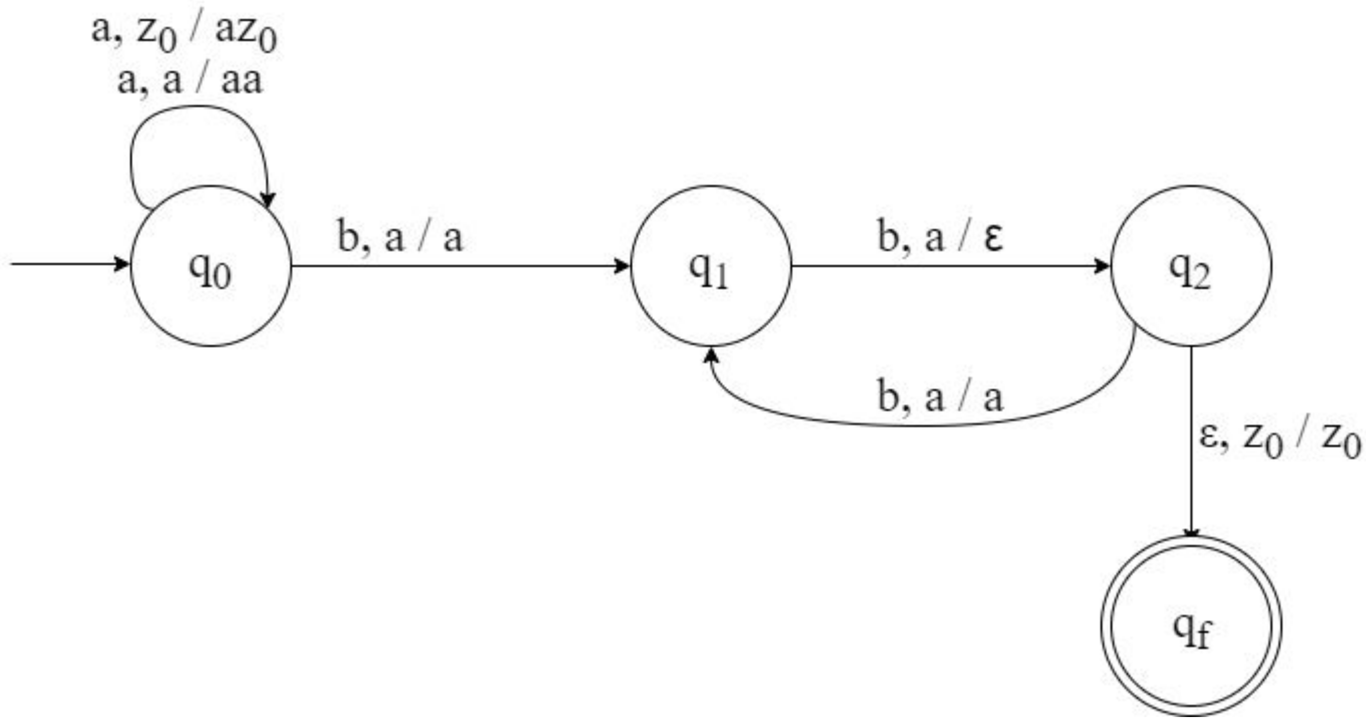
$\delta(q_0, b, a) = (q_1, \varepsilon)$

$\delta(q_1, b, a) = (q_1, \varepsilon)$

$\delta(q_1, \varepsilon, z_0) = (q_f, z_0)$

Design a PDA for the language $L=\{a^n b^{2n}, \quad n\geq 0\}$

a	**a**	**b**	**b**	**b**	**b**	**ε**



a
a
z_0

Design a PDA for the language $L=\{a^n b^{2n}, \quad n \geq 0\}$

$\delta(q_0, a, z_0) = (q_0, az_0)$

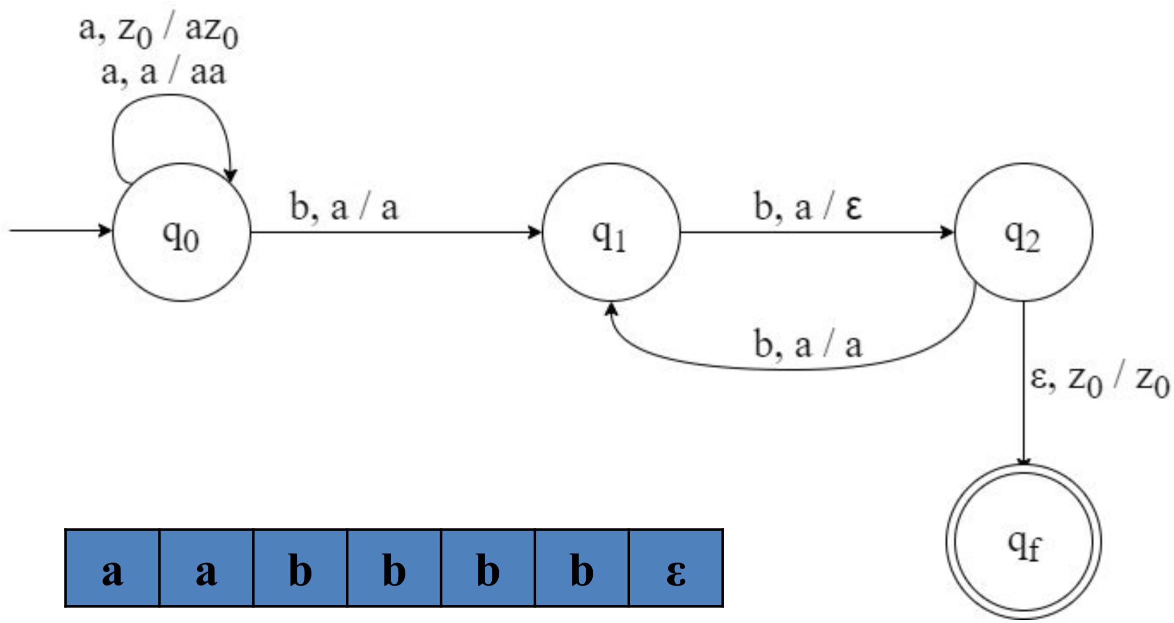
$\delta(q_0, a, a) = (q_0, aa)$

$\delta(q_0, b, a) = (q_1, a)$

$\delta(q_1, b, a) = (q_2, \varepsilon)$

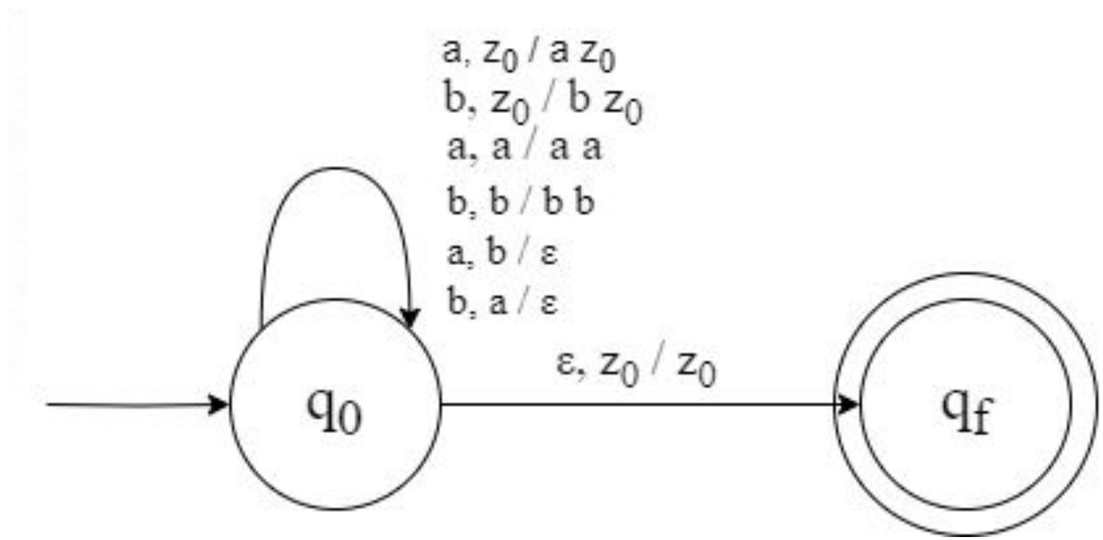
$\delta(q_2, b, a) = (q_1, a)$

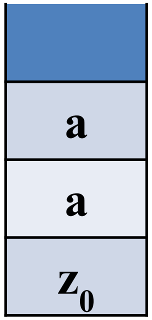
$\delta(q_1, \varepsilon, z_0) = (q_f, z_0)$



Design a PDA for the language $L=\{w \in (a, b)^* : n_a(w) = n_b(w)\}$

a	**a**	**b**	**b**	**ε**



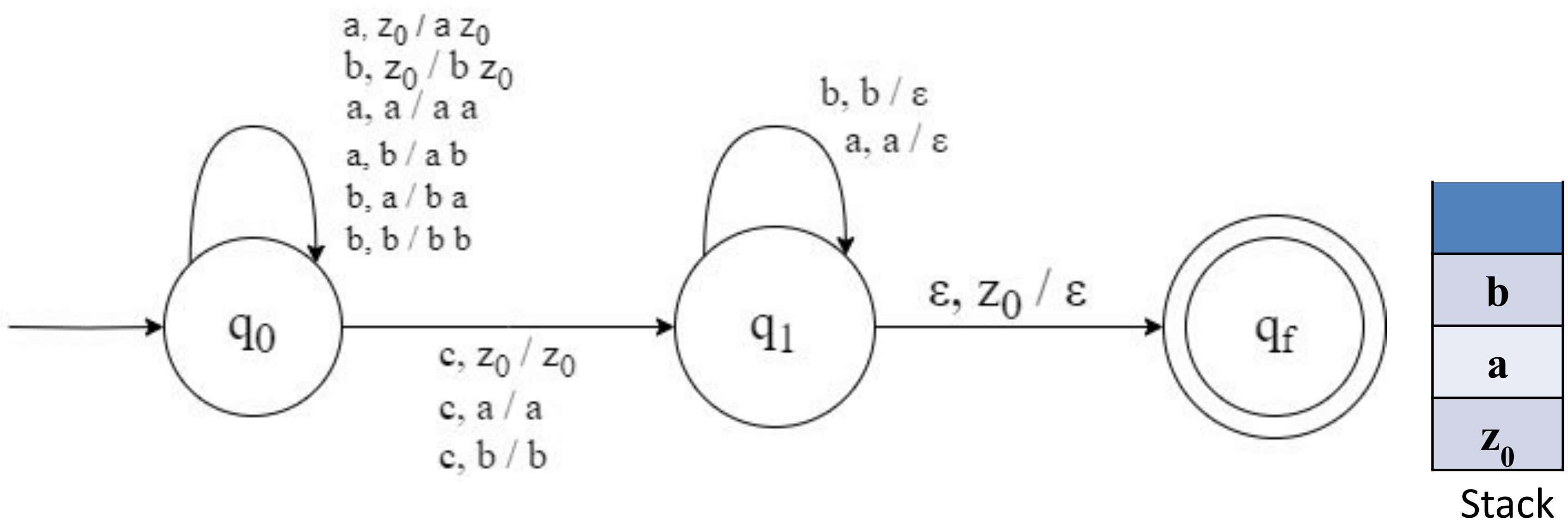


Stack

Design a PDA for the language $L=\{wcw^R, w \in (a, b)^*\}$

Input tape

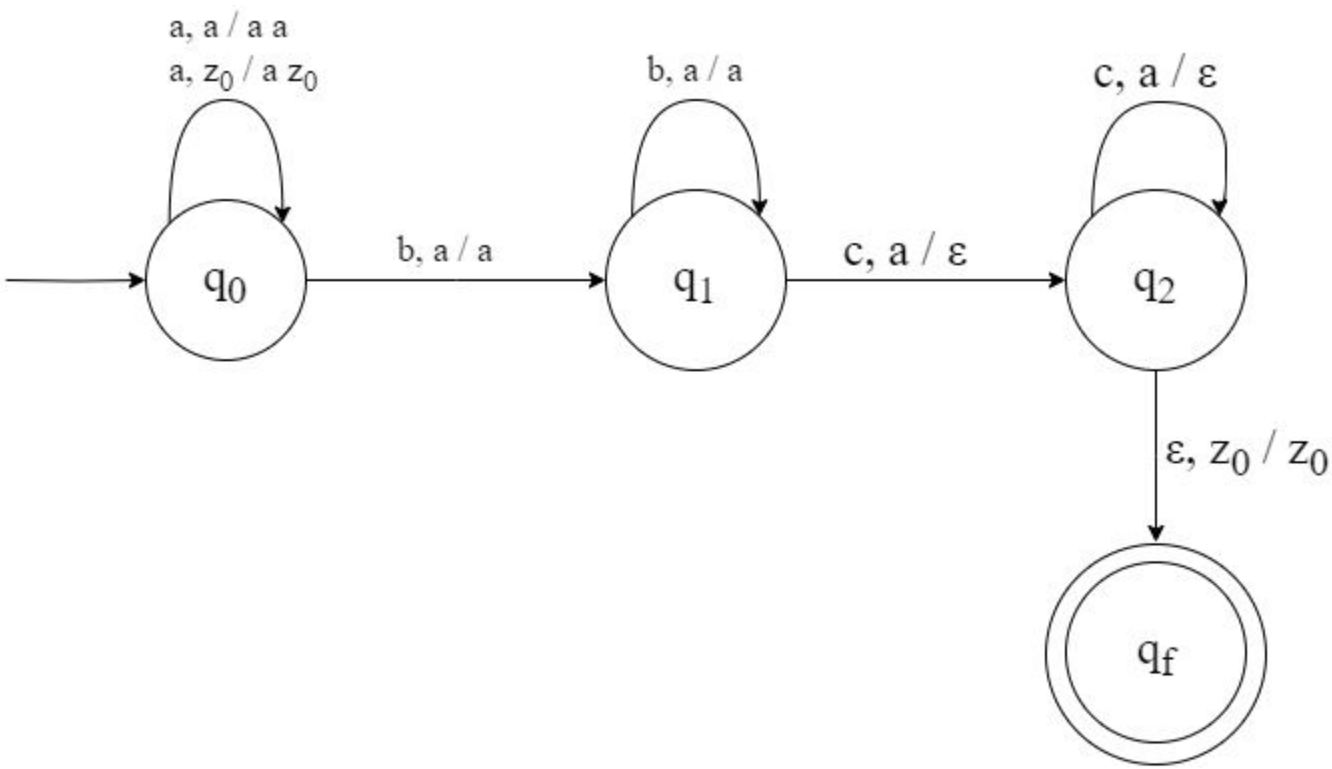
a	**b**	**c**	**b**	**a**	ε



Design a PDA for the language $L=\{a^n b^m c^n, n, m \geq 1\}$

Input tape

a	a	b	c	c	ε



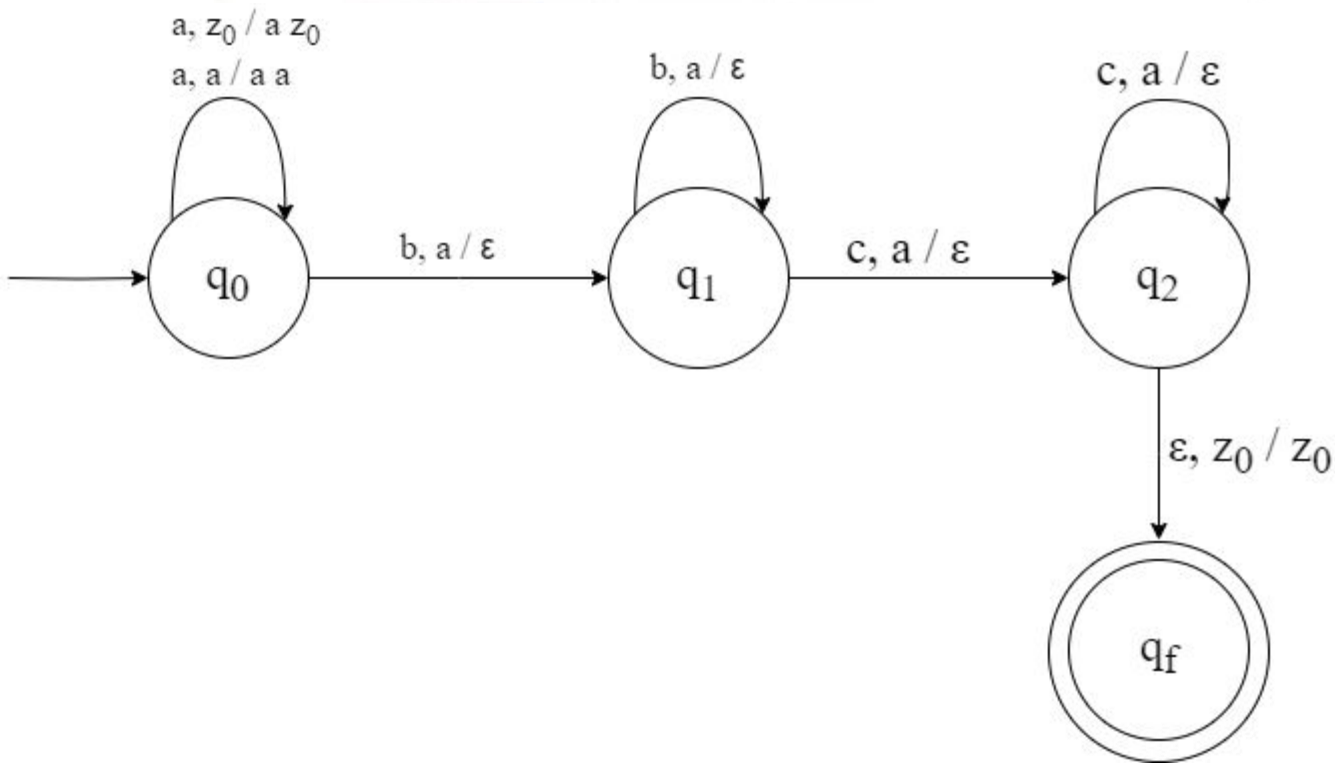
a
a
z_0

Stack

Design a PDA for the language $L=\{a^{m+n}b^mc^n,\ m, n\geq 1\}$

Input tape

a	**a**	**a**	**b**	**c**	**c**	**ε**



a
a
a
z_0

Stack

Practice problem

Design a PDA for the language $L=\{a^n b^m c^m d^n \; m, n \geq 1\}$

Suggested readings

1. An introduction to FORMAL LANGUAGES and AUTOMATA by PETER LINZ.
2. Introduction to Automata Theory, Languages, And Computation by JOHN E. HOPCROFT, RAJEEV MOTWANI, JEFFREY D. ULLMAN
3. Theory of computer science: automata, languages and computation **by** K.L.P MISHRA

Thank you