# Theoretical Computer Science

## Lab Session 7

March 18, 2021

# Agenda

- PDT - Push-Down Transducer
- Midterm Review session

PD transducer.

# PDT – Formal Definition

## A Pushdown Transducer

A PDT is a tuple $\langle Q, I, \Gamma, \delta, q_0, Z_0, F, O, \eta \rangle$ where

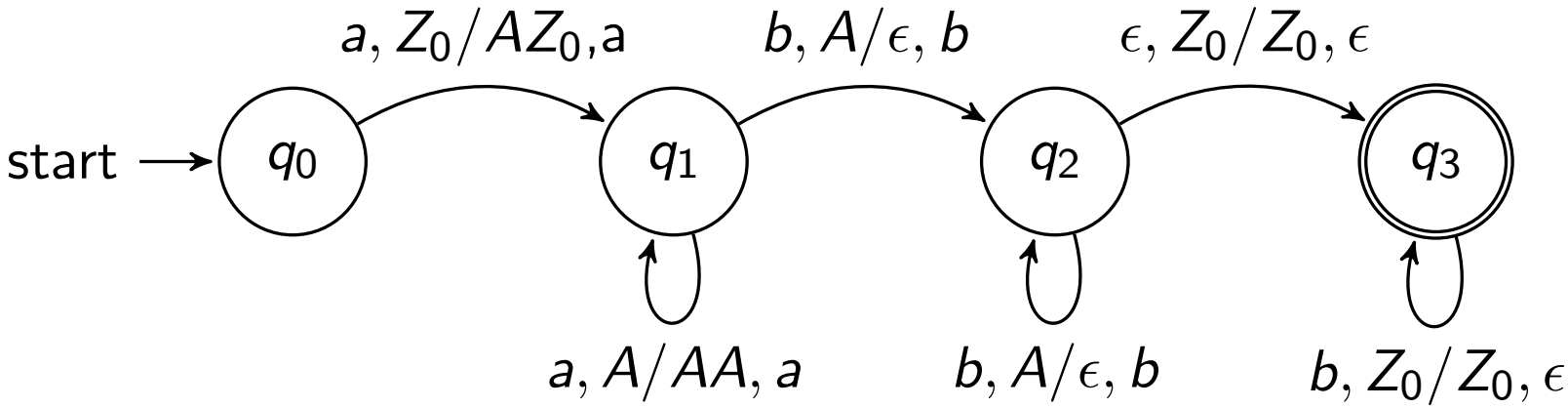
- $Q$ is a finite set of states.
- $I$ and $\Gamma$ are finite sets, the input and stack alphabets.
- $\delta$, the transition function, is a partial function from $Q \times (I \cup \{\epsilon\}) \times \Gamma$ to the set of finite subsets of $Q \times \Gamma^*$.
- $q_0 \in Q$, the initial state.
- $Z_0 \in \Gamma$, the initial stack symbol.
- $F \subseteq Q$, the set of accepting states.
- O is the output alphabet
- $\eta : Q \times (I \cup \{\epsilon\}) \times \Gamma \to O^*$

# Exercises

- 1. Build PDT that accepts x $\in L_1$, where $L_1 = \{a^n b^m | n \geq 1 \wedge n \leq m\}$, and translates it into $a^n b^n$
- 2. Build PDT that recognises $L_2 = \{xc | x \in \{a, b\}^+\}$ and reverses it

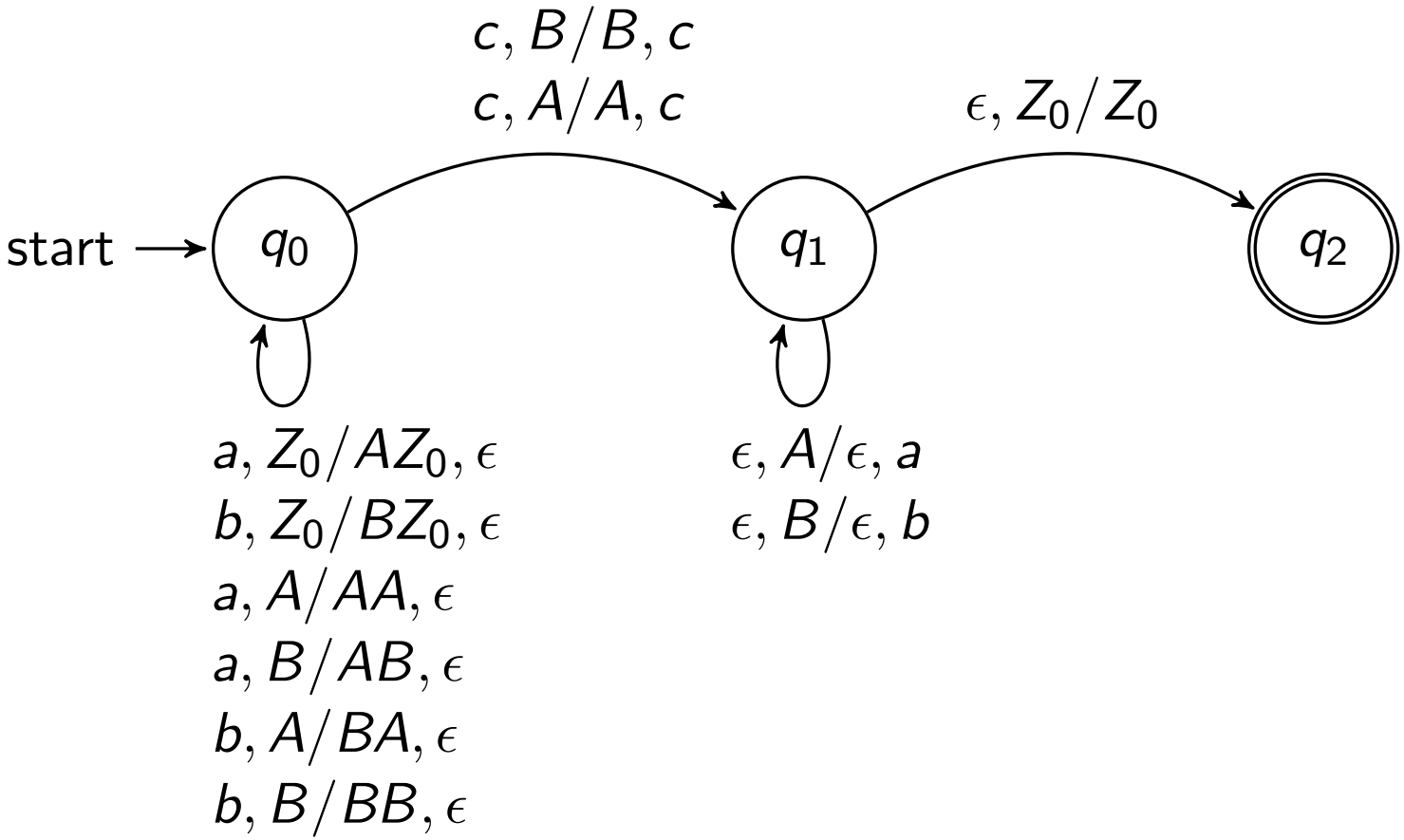
# Solution (1)

PDT that accepts x $\in L_1$, where $L_1 = \{a^n b^m | n \geq 1 \wedge n \leq m\}$, and translates it into $a^n b^n$



# Solution (2)

2. PDT that recognises $L_2 = \{xc | x \in \{a, b\}^+\}$ and reverses it



# Homework

Consider the language of well-formed parentheses of the arithmetic expressions (binary operations). Examples of strings belonging to the language are:

- (a + b)
- ((a) + (b + c))
- ((a + b))

Build PDT that recognises this language and translates it into reverse polish notation. For simplicity, consider the following alphabet $I = \{a, (, ), +\}$ – a single symbol ('a') that represents terms 'a', 'b', 'c', .... And a single operator ('+').

# Extra Exercises on DPDA and DPDT

Build DPDT that accepts the following languages

- $L_1 = \{a^n b^m a^n | m, n \in \mathbb{N} \text{ and } n \geq 1\}$, and translates it into $a^n b^m$
- $L_2 = \{a^i b^j c^k | i, j, k \in \mathbb{N} \text{ and } i + k = j\}$, and translates it into $a^i b^i c^k$
- $L_3 = \{xcy \mid x, y \in \{a, b\}^* \wedge |x| > 0 \wedge |y| > 0 \wedge y \neq x^R\}$ (where $x^R$ is the reversed string $x$), the alphabet is $I = \{a, b, c\}$ , and translates it into y
- $L_4 = \{a^n b^m | m, n \in \mathbb{N} \text{ and } n \leq m \leq 2n\}$, and translates it into $a^n b^n$

# Midterm review

- Tasks 1, 2 were graded by testing JFLAP solutions. The grade equals to 5-N, where N is the number of failed tests. In case if JFLAP implementation was different from the paper submission, it received 0 points (although we tried not to penalize for typos).
- The solutions of task1 obtained by swapping the initial and final state in the given incomplete automaton received 0 points and are not a subject for a review.
- Task 3 was graded manually according to the paper solution. The correct solution should accept the empty language and reflect that you understand how the operations on languages are performed. The cases when the operations were correctly performed on incorrect automata might be a subject for a review to be granted a partial grade.

**Please keep in mind that during the review the grade might change in any direction.**