

AUTOMATA FORMAL LANGUAGES AND LOGIC



Lecture notes on Non Deterministic Pushdown Automata

**Prepared by:
Prof.Divya S J
Assistant Professor**

Department of Computer Science & Engineering

PES UNIVERSITY

**(Established under Karnataka Act No.16 of 2013)
100-ft Ring Road, BSK III Stage, Bangalore - 560 085**

Table of contents

Section	Topic	Page number
1	Non Deterministic Pushdown Automata	3

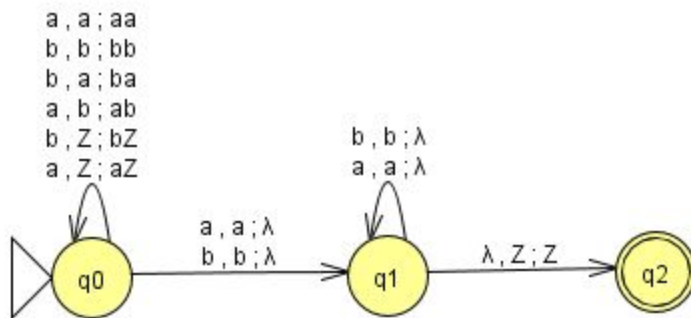
Examples Solved

#	Constructing Linear Grammar	Page number
1	Construct a NPDA for $L = \{ww^R \mid w \in \{a,b\}^*\}$	3
2	Construct a NPDA for $L = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$	4
3	Construct a NPDA for $L = \{a^i b^j c^k d^l \mid i=k \text{ or } j=l\}$	5

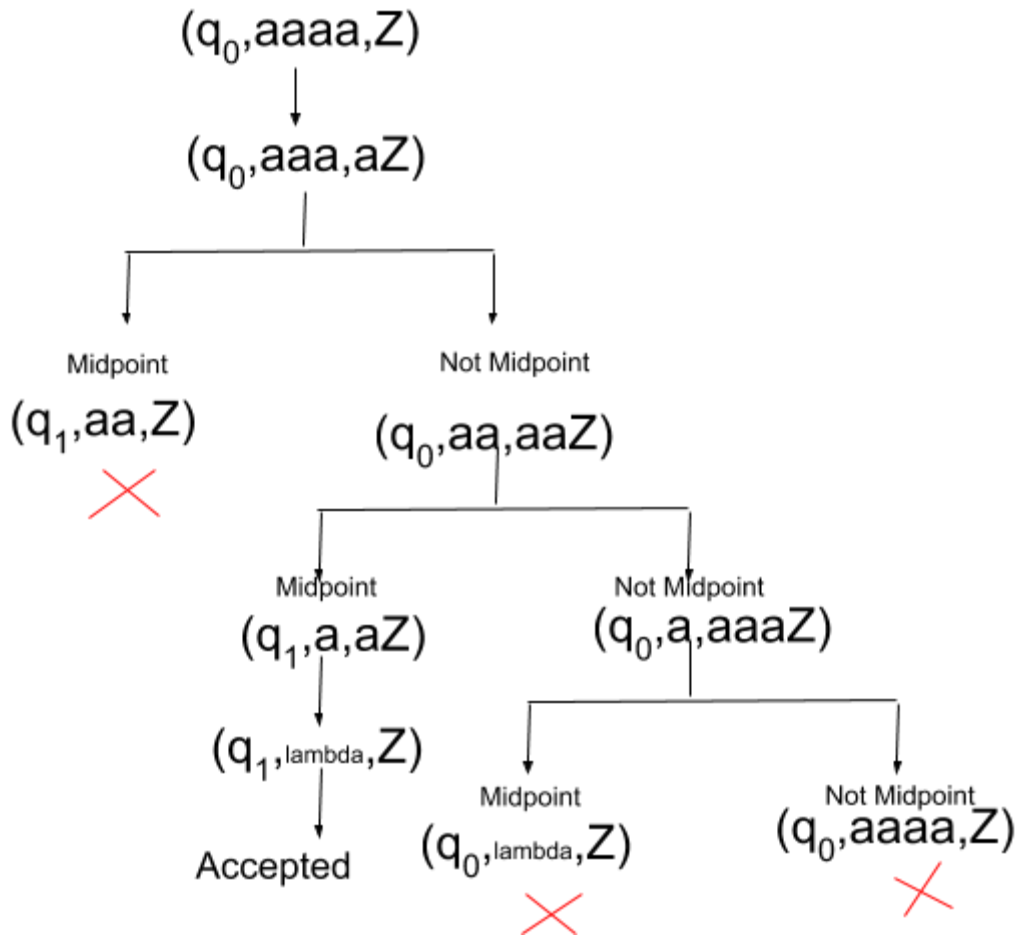
Non Deterministic Pushdown Automata

Let us consider language $L = \{ww \mid w \in \{a,b\}^*\}$ which is an even palindrome. The PDA however, is required to process the string left to right in a single pass. As such, a difficulty arises in processing an input that is a potential even palindrome: where is the midpoint of the input? The automaton cannot look ahead till the end of the input to determine the midpoint.

The PDA solves this problem by non-determinism the same way as NFAs. In its start state q_0 , it consumes the first half of the input string, remembering all of it by a's and b's onto the stack, no matter in which order they appear. It guesses the midpoint non deterministically and jumps to state q_1 , without consuming any input symbol and irrespective of what is at the top of the stack at that point.

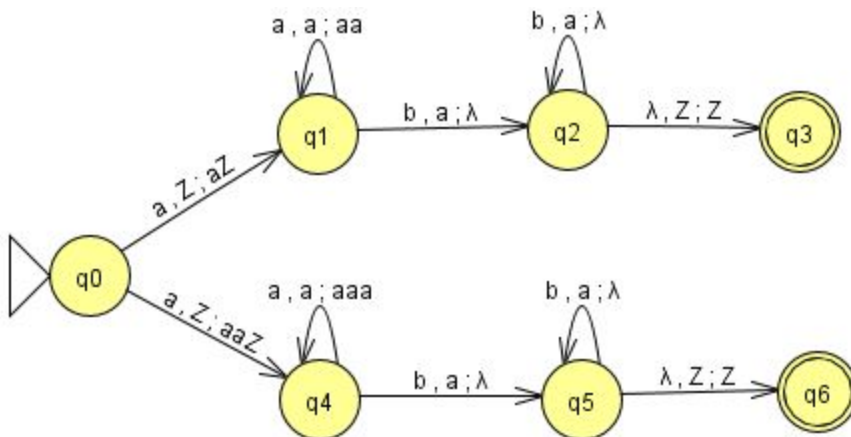


The configuration for the string aaaa is



$$2. L = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$$

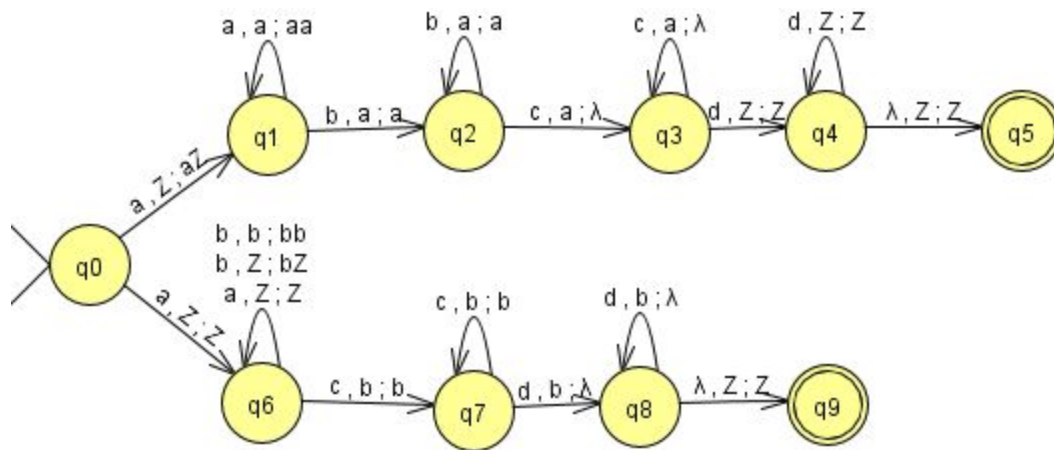
Solution:



$$3. L = \{a^i b^j c^k d^l \mid i=k \text{ or } j=l, i, j, k \geq 1\}$$

Solution: The above language can be written as $L = \{a^m b^j c^m d^l\} \cup \{a^i b^n c^k d^n\}$. The PDA has a nondeterministic branch at q_0 . If the string is $a^i b^j c^k$ with $i = j$, then the

PDA takes the branch from q_0 to q_1 . If the string is $a^j b^j c^j$ with $j = k$, then the PDA takes the branch from q_0 to q_6 .



DPDAs are less powerful than NPDAs.

- If we try to capture all different configurations in machine, NPDA could be at the same time in an equivalent DPDA, there will be multiple configurations of stack to be modelled.
- Stack contents are different in each parallel execution.
- These multiple versions of stack cannot be modelled in a single stack of a DPDA.

DPDA can recognize Deterministic Context Free Languages only. NPDA can recognize all CFLs.

An equivalent NPDA can be constructed for every DPDA.