Theory of Automata and Formal Language

Lecture-33

Dharmendra Kumar
(Associate Professor)
Department of Computer Science and Engineering United College of Engineering and Research, Prayagraj March 30, 2021

Construction of CFG from given PDA

Procedure:

Suppose the given PDA is $M = (Q, \Sigma, \Gamma, \delta, q_o, Z_o, \phi)$.

The context free grammar equivalent to this PDA is constructed as following:-

$G = (V, \Sigma, S, P)$, Where

$V = \{S\} \cup \{ [p, Z, q] \,!\, p, q \in Q \text{ and } Z \in \Gamma \}$

And P is defined by following three types of rules:-

1) Add $S \rightarrow [q_0, Z_0, p]$ into P, for every $p \in Q$.
2) If $(q, \epsilon) \in \delta(p, a, Z)$ then add $[p, Z, q] \rightarrow a$ into P for every $p, q \in Q$, $a \in (\Sigma \cup \{\epsilon\})$ and $Z \in \Gamma$.
3) If $(q, A_1 A_2 A_3 \ldots \ldots An) \in \delta(p, a, Z)$ then add

$[p, Z, q_1] \rightarrow a[q, A_1, q_2][q_2, A_2, q_3]\ldots\ldots\ldots\ldots\ldots[q_n, A_n, q_1]$

into P for every $p, q_i \in Q$, $a \in (\Sigma \cup \{\epsilon\})$ and $Z, Ai \in \Gamma$.

Where $1 \leq i \leq n$.

Ex. Construct CFG equivalent to the following PDA:-

$$M = (\{q_o, q_1\}, \{a, b\}, \{A, Z_o\}, \delta, q_o, Z_o, \phi)$$

And δ is defined as following:-

$\delta(q_o, a, Z_o) = \{(q_o, AZ_o)\}$ $\delta(q_o, a, A) = \{(q_o, AA)\}$

$\delta(q_o, b, A) = \{(q_1, A)\}$ $\delta(q_1, b, A) = \{(q_1, A)\}$

$\delta(q_1, a, A) = \{(q_1, \epsilon)\}$ $\delta(q_1, \epsilon, Z_o) = \{(q_1, \epsilon)\}$

Solution:

The equivalent CFG is constructed as

$G = (V, \Sigma, S, P)$, Where

$V = \{ S, [q_0, Z_0, q_0], [q_0, Z_0, q_1], [q_0, A, q_0], [q_0, A, q_1],$
$[q_1, Z_0, q_0], [q_1, Z_0, q_1], [q_1, A, q_0], [q_1, A, q_1] \}$

$\Sigma = \{a, b\}$

And P is determined as following:-

Type-1:$S \rightarrow [q_0, Z_0, q_0] / [q_0, Z_0, q_1]$

Type-2: In this type, we consider only the transition rules which pop the symbols.

Consider rule, $\delta(q_1, a, A) = \{(q_1, \epsilon)\}$

The production rule for it will be

$[q_1, A, q_1] \rightarrow a$

Similarly, for $\delta(q_1, \epsilon, Z_0) = \{(q_1, \epsilon)\}$

The production will be

$[q_1, Z_0, q_1] \rightarrow \epsilon$

Type-3: In this type, we consider only the transition rules which push the symbols or no push and no pop operation.

Consider rule, $\delta(q_0, a, Z_0) = \{(q_0, AZ_0)\}$

The production rule for it will be

$[q_0, Z_0, q_0] \rightarrow a\, [q_0, A, q_0]\, [q_0, Z_0, q_0]$

$[q_0, Z_0, q_0] \rightarrow a\, [q_0, A, q_1]\, [q_1, Z_0, q_0]$

$[q_0, Z_0, q_1] \rightarrow a\, [q_0, A, q_0]\, [q_0, Z_0, q_1]$

$[q_0, Z_0, q_1] \rightarrow a\, [q_0, A, q_1]\, [q_1, Z_0, q_1]$

Consider rule, $\delta(q_0, b, A) = \{(q_1, A)\}$

The production rule for it will be

$[q_0, A, q_0] \rightarrow b\, [q_1, A, q_0]$

$[q_0, A, q_1] \rightarrow b\, [q_1, A, q_1]$

Consider rule, $\delta(q_0, a, A) = \{(q_0, AA)\}$

The production rule for it will be

$[q_0, A, q_0] \rightarrow a\ [q_0, A, q_0]\ [q_0, A, q_0]$

$[q_0, A, q_0] \rightarrow a\ [q_0, A, q_1]\ [q_1, A, q_0]$

$[q_0, A, q_1] \rightarrow a\ [q_0, A, q_0]\ [q_0, A, q_1]$

$[q_0, A, q_1] \rightarrow a\ [q_0, A, q_1]\ [q_1, A, q_1]$

Consider rule, $\delta(q_1, b, A) = \{(q_1, A)\}$

The production rule for it will be

$[q_1, A, q_0] \rightarrow b\ [q_1, A, q_0]$

$[q_1, A, q_1] \rightarrow b\ [q_1, A, q_1]$

Ex. Construct CFG equivalent to the following PDA:-

$M = (\{q_o, q_1\}, \{a, b\}, \{A, Z_o\}, \delta, q_o, Z_o, \phi)$

And δ is defined as following:-

$\delta(q_o, a, Z_o) = \{(q_o, AZ_o)\}$ $\delta(q_o, a, A) = \{(q_o, AA)\}$

$\delta(q_o, b, A) = \{(q_1, \epsilon)\}$ $\delta(q_1, b, A) = \{(q_1, \epsilon)\}$

$\delta(q_1, \epsilon, A) = \{(q_1, \epsilon)\}$ $\delta(q_1, \epsilon, Z_o) = \{(q_1, \epsilon)\}$

Two Stack PDA(2PDA)

Two stack pushdown automata is described by a 7-tuple $M=(Q, \Sigma, \Gamma, \delta, q_o, Z_o, F)$ where,

- Q is the finite set of states,
- Σ is the set of input symbols
- Γ is the set of stack symbols,
- $q_o \in Q$ is the initial state,
- $Z_o \in \Gamma$ is a bottom symbol of stack
- F is the set of final states, and
- δ is a transition function which is defined as following:-
- $\delta: Q\times(\Sigma\cup\{\epsilon\})\times\Gamma\times\Gamma \rightarrow$ finite subset of $Q\times\Gamma^*\times\Gamma^*$

Ex. Construct 2-stack PDA for the following language $L = \{ a^n b^n c^n ! n \geq 1\}$.

Solution:

In this set, some strings are abc, $a^2b^2c^2$, $a^3b^3c^3$ etc.

Clearly, this set contains all the strings a, b and c, in which number of a, b and c are equal. And the order of a ,b and c are also fixed.

Procedure: In this PDA, we have to push the symbol A into first stack when a appears in input string.

When first b appears then we have to push symbol A into second stack and also change the state. For remaining b, we have to push symbol A into second stack at that state.

When first c appears then we check the top symbols of both stack. If both top symbols are A then we pop the top symbols from both stack. For remaining c, same operation is applied.

When string becomes empty, we check both stack. If top symbols of both stack are Zo, then string will be accepted.

Therefore the 2PDA for this language will be

$M = (\{q_{o,}\ q_1, q2,\}, \{a, b, c\}, \{A, Z_o\}, \delta, q_o, Z_{o,}\ \phi)$

And δ is defined as following:-

$\delta(q_o, a, Z_o, Z_o) = \{(q_o, AZ_{o,}, Z_o)\}$

$\delta(q_o, a, A, Z_o) = \{(q_o, AA_{,}, Z_o)\}$

$\delta(q_o, b, A, Z_o) = \{(q_1, A_{,}, AZ_o)\}$

$\delta(q_1, b, A, A) = \{(q_1, A_{,}, AA)\}$

$\delta(q_1, c, A, A) = \{(q_2, \epsilon, \epsilon)\}$

$\delta(q_2, c, A, A) = \{(q_2, \epsilon, \epsilon)\}$

$\delta(q_2, c, Z_o, Z_o) = \{(q_2, \epsilon, \epsilon)\}$

Transition diagram this 2PDA is