@ktunotes.in

GET IT ON Google Play

KTU NOTES

The learning companion.

KTU STUDY MATERIALS | SYLLABUS | LIVE NOTIFICATIONS | SOLVED QUESTION PAPERS

Website: www.ktunotes.in

31/01/22

Module 4

More on Context Free Language

Push Down Automata

RL ← RG (generates)
↑ recognises
FA

- → DPDA - Deterministic PDA
- → NPDA - Non-deterministic PDA

CFL ← CFG (generates)
↑ recognises
PDA

NPDA is powerful than DPDA.
(DPDA used for parsing).

∃ languages which can only be recognised by NPDA

PDA = ε - NFA + stack (memory)

PDA = ε-NFA + stack

FA → 5 tuples   PDA → 7 tuples

Ktunotes.in

Formal Description of NPDA

→ It is a 7-tuples.

→ $(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

Q - Finite non-empty set of states

Σ - Alphabet

Γ - Stack Alphabet

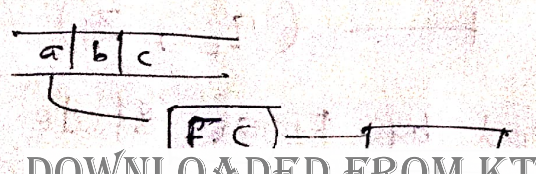
Γ - capital gamma.

δ - transition function

q_0 - start state

Z_0 - initial stack symbol

F - $Q \subseteq F$ set of final states



DOWNLOADED FROM KTUNOTES.IN

* $(p, a, X) \vdash (q, w)$

p ↑ current state; a ↓ current input; X ↓ stack top

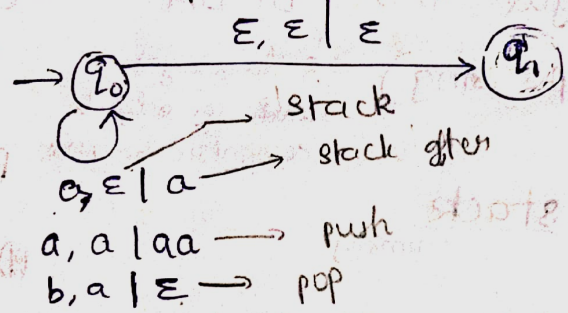
$\vdash$ Turnstile notation

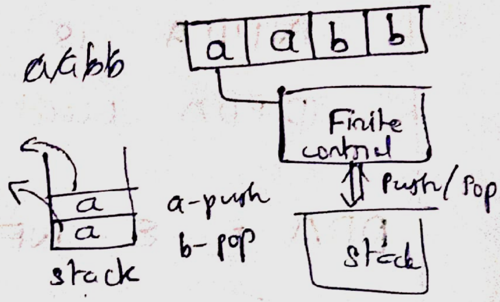
q ↓ state to which PDA progresses when consuming a

w ↪ stack after

} Transition in PDA

Q $L = \{a^n b^n \mid n > 0\}$





If stack is empty, we consider it belongs to the language. Acceptance by empty stack.

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

⇓

Transition function for PDA.

$\delta(q_0, \varepsilon, \varepsilon) \vdash (q_0, Z_0)$

$\delta(q_0, a, Z_0) \vdash (q_0, XZ_0)$

$\delta(q_0, a, X) \vdash (q_0, XXZ_0)$

⋮

$\delta(q_0, b, X) \vdash (q_1, \varepsilon)$ ⇒ Replace stack top with ε

$\delta(q_1, b, X) \vdash (q_1, \varepsilon)$

$\delta(q_1, \varepsilon, Z_0) \vdash (q_f, Z_0)$

→ accepted by Final State.

DOWNLOADED FROM KTUNOTES.IN

* For acceptance by final state, final state ⊚ is required, but in acceptance by empty stack final state is not required.

* NPDA can recognize any CFL, DPDA recognizes only DCFL (Deterministic CFL)

* For NPDA : $\delta : Q \times \Sigma \times T \longrightarrow Q \times T$ For any NPDA (Q can be replaced by 2^Q or $P(Q)$

* Instantaneous Description

Description of machine at any point of time

$(q, a, w) \vdash (p, x)$

$q, p \in Q$

$q \xrightarrow{a,\ w|x} p$

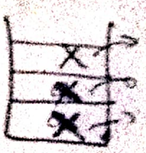
q → current state
a → current symbol
w → current stack top
p → next state
x → stack after push / pop

* Language Acceptance by a PDA

* Acceptance by Empty Stack
* Acceptance by Final State

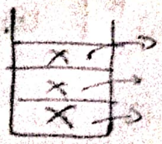
eg. $L = \{a^n b^n \mid n > 0\}$

aaabbb



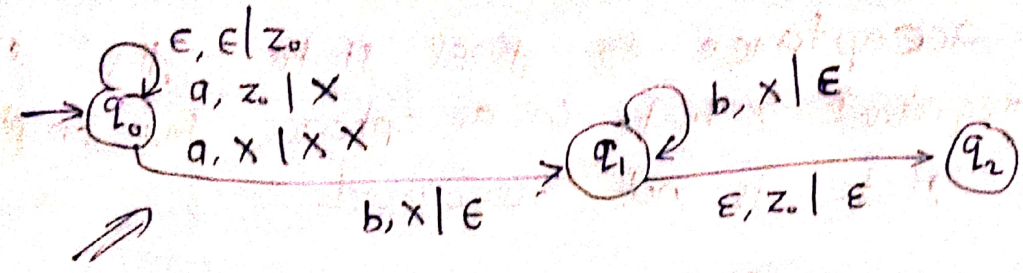
Empty stack at last so accep[ted]

·aaabbbb



Empty stack but still 1 b remains so not accepted.

DOWNLOADED FROM KTUNOTES.IN

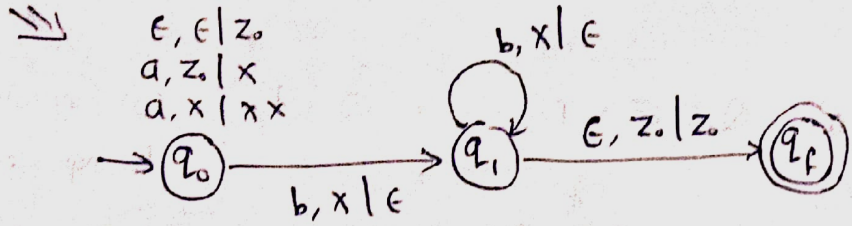


This is a DPDA.

⇒ Acceptance by Empty stack

Acceptance by Final state

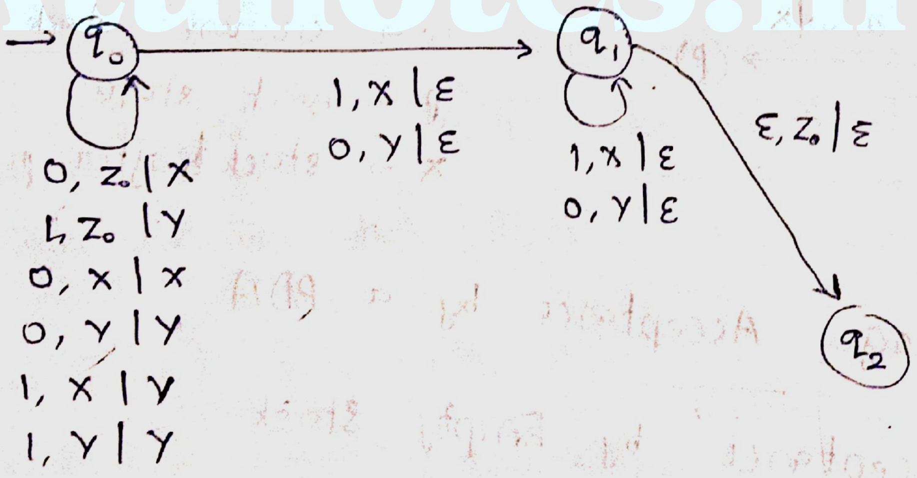
⇒

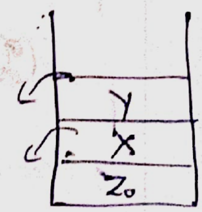


$L = \{ww \mid w \in \{0,1\}^*\}$

This DPDA has an issue, if $w = 01$, $ww = 0101$, it is hard to find out the boundaries of w. 01 01

∴ NPDA





Pop x for 1
Pop y for 0

(Non deterministic NPDA)

So we just assume that every element as a boundary 0 1 0 1 .

$L = \{wcw^R \mid w \in \{0,1\}^*\} \Rightarrow$ DPDA, ∵ Boundary known

$L = \{ww^R \mid w \in \{0,1\}^*\}$ (Palindrome)

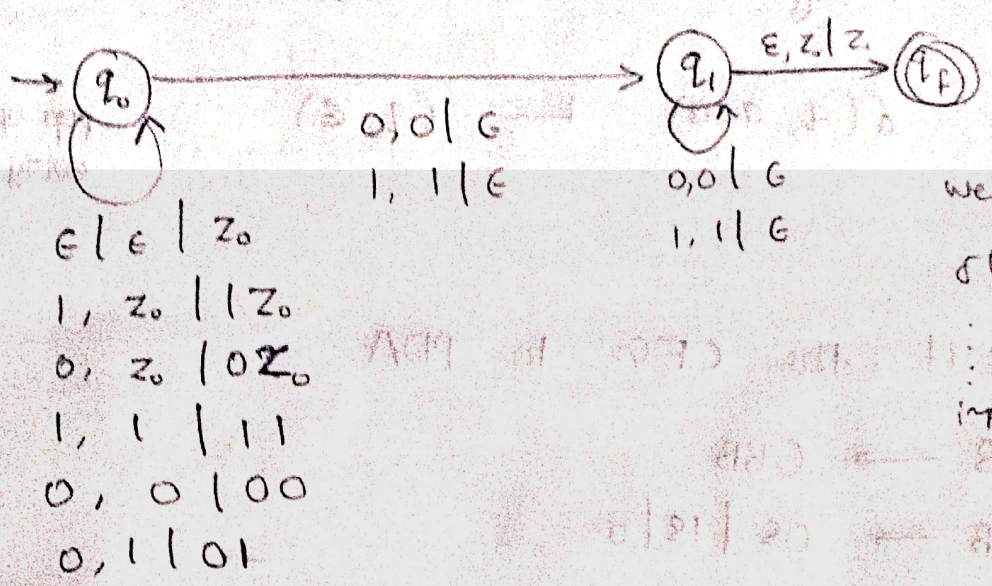
$w = 100$, $w^R = 001$ $ww^R = 100001$

Pushing x on 0, pop x on 0 in reverse

" Y on [illegible] pop Y [illegible] reverse

DOWNLOADED FROM KTUNOTES.IN

$L = \{ww^R \mid w \in \{0,1\}^*\}$



we have to mention

$\delta(q_0, \varepsilon, \varepsilon) \vdash \varepsilon, z_0$

important.

* Equivalence of CFG and PDA ($CFG \cong PDA$)

(1) conversion of CFG to PDA

Let $L = L(G)$, L is the language generated by the grammar G where $G = (V, T, P, S)$ where G is a CFG, we can construct a PDA A such that

$$A = (q, \Sigma, (V \cup \Sigma), \delta, q, S, \phi)$$

This is done by 2 rules -:

Rule 1 - If you have a production

$A \longrightarrow \alpha$ where A is a Non-terminal

then you add a transition,

$$\delta(q, \varepsilon, A) \longmapsto (q, \alpha)$$

DOWNLOADED FROM KTUNOTES.IN

Rule 2 : For every $a \in T$ (terminal in the gramm
we can define a transition,

$$\delta(q, a, a) \vdash (q, \epsilon)$$

pop operation for every terminal.

Q Convert the CFG to PDA.

$S \rightarrow 0BB$

$B \rightarrow 0S \mid 1S \mid 0$

Ans CFG should be in GNF.

This is in GNF.

Step 1 : Add this to stack top.

$\delta(q, \epsilon, \epsilon) \vdash (q, S)$

Rule 1:

$\delta(q, \epsilon, S) \vdash (q, 0BB)$ $\quad A \rightarrow \alpha$

$\delta(q, \epsilon, B) \vdash (q, 0S)$ $\quad S \rightarrow 0BB$

$\delta(q, \epsilon, B) \vdash (q, 1S)$ $\quad B \rightarrow 0S$

$\delta(q, \epsilon, B) \vdash (q, 0)$ $\quad B \rightarrow 1S$

$\quad B \rightarrow 0$

Rule 2:

$\delta(q, 0, 0) \vdash (q, \epsilon)$

$\delta(q, 1, 1) \vdash (q, \epsilon)$

Ktunotes.in

DOWNLOADED FROM KTUNOTES.IN

Conversion of PDA to CFG

Let PDA $P = (Q, \Sigma, T, \delta, q_0, Z_0)$

Going to construct a CFG, $G = (V, \Sigma, P, S)$

Variables

1. Start symbol S
2. $[p\, x\, q]$ where p and q are states of P $(p, q \in Q)$ and $x \in T^*$

Productions

1. For all states p $(\forall p \in Q)$

$$S \longrightarrow [q_0 Z_0 p]$$

$Q = \{q_0, q_1\}$
$[q_0 Z_0 q_0]$
$[q_0 Z_0 q_1]$

2. For all non-popping transitions,

Let $\delta(q, a, X) \longrightarrow (r, Z_1 Z_2 \ldots Z_n)$

where $Z_1 Z_2 \ldots Z_n$ are.

This will give rise to production

$$[q\, x\, r_n] \longrightarrow a[r Z_1 r_1][r_1 Z_2 r_2] \ldots [r_{n-1} Z_n r$$

$$r_n, r_1, r_2 \ldots \in Q$$

3. For popping instructions

$$\delta(q, a, x) \longrightarrow (r, \varepsilon)$$ where q, r a member of Q

$$[q\, x\, r] \longrightarrow a$$

$$\delta(q, \varepsilon, x)^{r} \longrightarrow (r, \varepsilon)$$

DOWNLOADED FROM KTUNOTES.IN

Q. construct a CFG for PDA $P = (\{q_0, q_1\}, \{0,1\}, \{X, Z\}, \delta, q_0, Z)$

Ans

(1) $\delta(q_0, 1, Z) \vdash (q_0, XZ)$

(2) $\delta(q_0, 1, X) \vdash (q_0, XX)$

(3) $\delta(q_0, \epsilon, X) \vdash (q, \epsilon)$

(4) $\delta(q_0, 0, X) \vdash (q_1, X)$

(5) $\delta(q_1, 1, X) \vdash (q_1, \epsilon)$

(6) $\delta(q_1, 0, Z) \vdash (q_0, Z)$

Rule 1.

$S \longrightarrow [q_0 Z q_0] \mid [q_0 Z q_1]$ —— I

Rule 2

Taking transition (1)

(1) $\delta(q_0, 1, Z) \vdash (q_0, XZ)$

$$\left.\begin{array}{l}
[q_0 Z q_0] \longrightarrow 1 [q_0 X q_0][q_0 Z q_0] \\
[q_0 Z q_0] \longrightarrow 1 [q_0 X q_1][q_1 Z q_0] \\
[q_0 Z q_1] \longrightarrow 1 [q_0 X q_0][q_0 Z q_1] \\
[q_0 Z q_1] \longrightarrow 1 [q_0 X q_1][q_1 Z q_1]
\end{array}\right\} \text{II}$$

$r_1, r_2 \in Q$

$Q = \{q_0 q_1\}$

(2) $\delta(q_0, 1, X) \vdash (q_0, XX)$

$$\left.\begin{array}{l}
[q_0 X q_0] \longrightarrow 1 [q_0 X q_0][q_0 X q_0] \\
[q_0 X q_0] \longrightarrow 1 [q_0 X q_1][q_1 X q_0] \\
[q_0 X q_1] \longrightarrow 1 [q_0 X q_0][q_0 X q_1] \\
\quad \longrightarrow 1 [q_0 X q_1][q_1 X q_1]
\end{array}\right\} \text{III}$$

DOWNLOADED FROM KTUNOTES.IN

③ $\delta(q_0, \epsilon, X) \vdash (q_0, \epsilon)$ popping instruction

$[q_0 X q_0] \rightarrow \epsilon$ —— (IV)

④ $\delta(q_0, 0, X) \vdash (q_1, X)$

$$\left.\begin{array}{l} [q_0 X q_0] \longrightarrow 0[q_1 X q_0] \\ [q_0 X q_1] \longrightarrow 0[q_1 X q_1] \end{array}\right\} \text{(V)}$$

⑤ $\delta(q_1, 1, X) \vdash (q_1, \epsilon)$

$[q_1 X q_1] \longrightarrow 1$ —— (VI)

⑥ $\delta(q_1, 0, Z) \longrightarrow (q_0, Z)$

$$\left.\begin{array}{l} [q_1 Z q_0] \longrightarrow 0[q_0 Z q_0] \\ [q_1 Z q_1] \longrightarrow 0[q_0 Z q_1] \end{array}\right\} \text{(VII)}$$

Pumping Lemma for CFL

It is used to prove that a language is not context free.

Statement:

If L is a context free language, then L has a pumping length 'p' such that any string 's' having length greater or equal [illegible] 1 can

DOWNLOADED FROM KTUNOTES.IN

be divided into 5 parts $s = uvxyz$ such that following conditions hold :

1) $|vy| > 0$
2) $|vxy| \leq p$
3) $uv^i x y^i z \in L$ for every $i \geq 0$

Steps (Proof by contradiction)

1. Assume that the language is CFL
2. Then it has a pumping length say p
3. All strings in L having length $\geq p$ can be pumped.
4. Find some string 's' in L such that $|s| \geq p$
5. Divide s into 5 parts

   $s = uvxyz$
6. Show that $uv^i xy^i z \notin L$ for some i
7. This contradicts our assumption that L is a CFL. Hence L is not CFL.

Eg. Show that $L = \{a^n b^n c^n \mid n \geq 0\}$ is not a CFL.

1. Assume that L is a CFL
2. L has a pumping length 'p'
3. Let s be a string in L having length $\geq p$

s =

DOWNLOADED FROM KTUNOTES.IN

Let us say $p = 6$

$$S = \underbrace{a}_{u}\underbrace{aa}_{v}\ \underbrace{a}_{x}\underbrace{aa}_{y}\ \underbrace{bbb\,bbb\,cccccc}_{z}$$

Case 1 :

$|vxy| \le p$

$|vy| > 0$

$uv^i x y^i z$

$$S = (a)(aa)^2(a)(aa)^2 b^6 c^6$$

$= a^{10} b^6 c^6 \notin L$ $\therefore$ pumps comes in 'a' part

$\therefore$ Our assumption is wrong

Case 2 :

$p = 6$

$\underbrace{aaaa}_{u}aa\ bbbbbb\ \underbrace{cccccc}_{z}$

$$S = a^4(a^2b)(bb)(b)(b^2c^6)$$

$= a^4 a^4 b^2 b^2 b^2 c^6$

$= a^8 b^8 c^6 \notin L$.

Case 3 : and case 4 we get the same.

Hence our assumption is contradicted

Ktunotes.in

DOWNLOADED FROM KTUNOTES.IN

* Closure properties of CFL

① Union

Let L_1 and L_2 be two CFL's. Then G_1 and G_2 be the CFG corresponding to L_1 and L_2.

Let

$G_1: S \rightarrow aS_1b \mid \epsilon$ $\quad$ $G_2: S_2 \rightarrow bS_2c \mid \epsilon$

$L_1: \{a^n b^n \mid n \geq 0\}$ $\quad$ $L_2: \{b^n c^n \mid n \geq 0\}$

$$G \Rightarrow \begin{bmatrix} S \rightarrow S_1 \mid S_2 \\ S_1 \rightarrow aS_1b \mid \epsilon \\ S_2 \rightarrow bS_2c \mid \epsilon \end{bmatrix} \Rightarrow CFL$$

If S_1 is corresponding to L_1 and S_2 is corresponding to L_2, if we introduce another symbol S, where S_1 and S_2 are derived from the new starting symbol, then you get CFG

CFG for L where $L = L_1 \cup L_2$

$\therefore$ $L = L_1 \cup L_2$ is a CFL.

② Contradiction

L_1 and L_2 are CFL

we get G_1 and G_2

$S \rightarrow S_1 . S_2$

$S_1 \rightarrow aS_1b \mid \epsilon$

$S_2 \rightarrow b$

DOWNLOADED FROM KTUNOTES.IN

can have $L_1 . L_2$ be a CFL because we can combine S_1 and S_2 with '.' for G ; which is a CFG.

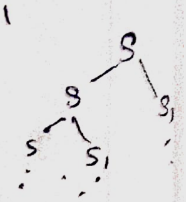
L(G) will be CFL.

③ Kleene Closure

L_1 is a CFL

$G_1 \xRightarrow{L} L_1$

whatever grammar G_1 can generate, we can find $(L_1)^*$ by introducing something to the starting. Let S_1 is start symbol for G_1



$S \longrightarrow SS_1 / \varepsilon$ (make it recursive)

∴ It is a CFG, Kleene closure of any CFL is closed.

④ Intersection and Complementation

The set of CFL is NOT CLOSED

Intersection

$L_1 = \{a^n b^n c^m \mid n \geq 0, m \geq 0, m > n\}$

$L_2 = \{a^m b^n c^n \mid n \geq$ " $\}$

$L_1 \cap L_2$

$= \{a^n b^n c^n \mid n \geq 0\}$ ≠ CFL . but CSL

DOWNLOADED FROM KTUNOTES.IN

complement

$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$

Assume the set of CFL is closed under compliment

∴ ~~$L_1 \cap L_2$ is closed~~, $\overline{L_1} \cup \overline{L_2}$ is closed, $\overline{CFL \cup CFL}$

∵ union is closed, $\overline{CFL}$ is closed = CFL

∴ But intersections are not closed ∴ $L_1 \cap L_2$ is not closed.

∴ Our assumption is not true. ∴ Compliment is also not closed.

Ktunotes.in

DOWNLOADED FROM KTUNOTES.IN