Name : Shubham Dutta

Year : 2nd

Stream : CST

Section : 2B

Roll : 58

Enrollment No. : 12019009022112

Paper Name : Formal Language & Automata Theory

Paper Code : PCC-CS401

Signature : Shubham Dutta

Date : 05/05/2021

Answers

1.A) $W_1 = (A, C)$ as $A \to a$ & $C \to b$ are

Step-1 productions with a ~~ter~~ terminal string on RHS

$W_2 = \{A, C\} \cup \{A_1 \mid A_1 \to \alpha$ with $\alpha \in (\Sigma \cup \{A, C\})^*\}$

$= \{A, C, S\} \cup \phi$

As $W_3 = W_2$

$V_N' = W_2 = \{S, A, C\}$

$P' = \{A_1 \to \alpha \mid A_1 ; \alpha \in (V_N' \cup \Sigma)^*\}$

Thus $= \{S \to CA, A \to \alpha, C \to b\}$

$G_1 = \{(S, A, C), \{a, b\}, \{S \to CA, A \to a, C \to b\}, S\}$

Step-2 We have to apply Theorem to G_1 Thus

$W_1 = \{S\}$

As we have production $S \to CA$ & $S \in W_1$, ~~W_2~~

$W_2 = \{S\} \cup \{A, C\}$

As $A \to a$ & $C \to b$ are productions with
$A, C \in W_2$, $W_3 = \{S, A, C, a, b\}$

As $W_3 = V_N' \cup \Sigma$,

$P'' = \{S \to \alpha \mid A, \in W_3\} = P'$

$\therefore$ ~~$G' = \{$~~

$\therefore G' = (\{S, A, C\}, \{a, b\}, \{S \to CA, A \to a, C \to b\}, S)$

~~2.A)~~

3 b) We have to start with an S-production. At every stage we apply a suitable production which is likely to derive w. ~~S~~

In this example, substring is underlined to be replaced by use of production

$$S \Rightarrow \underline{aA_1} A_2 a$$
$$\Rightarrow baa \underline{A_2} a$$
$$\Rightarrow baa \underline{A_1} aba$$
$$\Rightarrow bahab \underline{aA_1} A_2 baba$$
$$\Rightarrow baabbaa \underline{A_2} baba$$

$\therefore S \Rightarrow baabba\, \underline{aA_1}\, abbaba$

$\Rightarrow baabbabaaabbaba = w$

$\therefore \boxed{w \in L(G)}$

2A) $S \to AB$, $A \to BS \mid b$, $B \to SA \mid a$

Then

$S \to AB$ is not in GNF. so substitute $A \to BS \mid b$ then we get,

$S \to BSB \mid bB$

However $S \to BSB$ is not in GNF, so substitute $B \to SA \mid a$ we get

$S \to SASB \mid aSB \mid bB$

Here, $S \to SASB$ is left recursion.

So, $C \to SA \mid SAC$.

then production becomes

$S \to aSBC \mid bBC \mid aSB \mid bB$

4. A) $L = \{a^n b^n \mid n \geq 1\}$.

$\therefore L = \{ab,$ ~~ab~~ $aabb, aaabbb,$ ~~aaaabb~~ $aaaabbbb, \dots\}$

As we want to design a PDA, then every time a comes before b, when 'a' comes then push it in stack & if again 'a' comes then also push it. After that, when 'b' comes then pop one 'a' from the stack each time. So, at the end if the stack becomes empty then we can say that string is accepted by PDA.

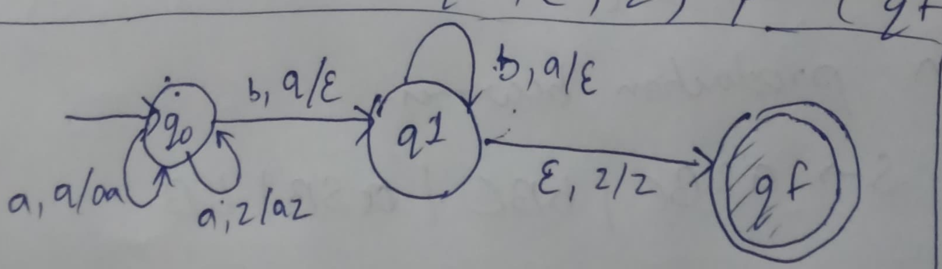
Stack transaction functions :-

$q_0 \rightarrow$ initial state

$q_f \rightarrow$ Final state

$\varepsilon \rightarrow$ indicates pop operation

$$\delta(q0, a, z) \vdash (q0, az)$$
$$\delta(q0, a, a) \vdash (q0, aa)$$
$$\delta(q0, b, a) \vdash (q1, \epsilon)$$
$$\delta(q1, b, a) \vdash (q1, \epsilon)$$
$$\delta(q1, \epsilon, z) \vdash (qf, z)$$



5. A) $S \to AMB$

$M \to aMb$

$M \to \varepsilon$

Removal of Null production

$M \to \varepsilon$

In RHS M is present

$S \to AMS$

$M \to aMb$

<u>Case 1</u> :-

$S \to AMB$

$S \to AB$ [$\because M \to \varepsilon$]

<u>Case 2</u> :-

$M \to aMb$

$M \to ab$ [$\because M \to \varepsilon$]

After removal of null production

$S \to AMB \mid AB$

$M \to aMb \mid ab$

6. A) $L = \{a^p \mid p \text{ is prime}\}$ is NOT CFL

There is no pattern for prime no. can't be identified by both FSA (Finite State Automation).

We just given the p = prime no. & there is no bound on the length. So, we need infinite amount of space which can be provided by Turing machine

$\therefore$ a^p where p is prime is a CSL

$w = xyz$

$|xyz| = p \qquad |y| \geq 1$

$|xy'z| = |xyz| + |y| p$

$= p + (|y|)$

7. B) Input

B	1	0	1	0	1	0	1	0	B

Output

B	0	1	0	1	0	1	0	1	B