Assignment 2

Q.1 Design NPDA for accepting the language WW^R where W belongs to $\{a,b\}^*$ and W^R is the reverse of string W.

→ Given : $L = \{WW^R \mid W \in \{a,b\}^*\}$

W^R is the reverse of string W
If W = ab then W^R = ba so
WW^R = abba

L = {aa, bb, abba, aabbaa, abaaba, ... }
NPDA can be described by 7 tuples

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

Q = set of states
Σ = Input alphabet
Γ = stack alphabet
δ = Transition function
q_0 = Initial state
Z_0 = Initial stack symbol
F = set of Final states

Stack transition Function:

$\delta(q_0, a, Z_0) \vdash (q_0, aZ_0)$

$\delta(q_0, a, a) \vdash (q_0, aa)$

$\delta(q_0, b, Z_0) \vdash (q_0, bZ_0)$

$\delta(q_0, b, b) \vdash (q_0, bb)$

$\delta(q_0, a, b) \vdash (q_0, ab)$

$\delta(q_0, b, a) \vdash (q_0, ba)$

$\delta(q_0, a, a) \vdash (q_1, \epsilon)$

$\delta(q_0, b, b) \vdash (q_1, \epsilon)$

$\delta(q_1, a, a) \vdash (q_1, \epsilon)$

$\delta(q_1, b, b) \vdash (q_1, \epsilon)$

$\delta(q_1, \epsilon, z_0) \vdash (q_F, z_0)$

where

$Q = \{q_0, q_1, q_F\}$

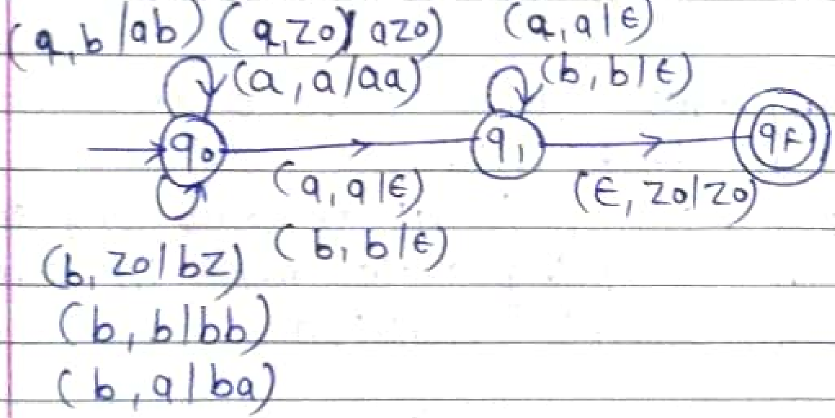
$q_0 =$ initial state

$z_0 =$ initial stack symbol

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

$F = \{q_F\}$

$\Gamma = \{a, b, z_0\}$



- when 'a' or 'b' comes then either push into stack or move into the next state.

- when the input alphabet which is equal to top of the stack then that time pop operation applies on stack and move to the next step.
- If stack becomes empty then we can say that the string is accepted by NPDA.

Q.2 write short note on recognition of language by PDA

→ 1) A pushdown automata (PDA) is type of automata used in automata theory for recognizing context free languages, which are languages that can be generated by context-free grammars.

2) PDA extends power of finite automata by including a stack as an additional memory structure allowing them to recognize languages that involve recursive and nested patterns.

3) Recognition of language by PDA.
- PDA processes input string symbol by symbol. maintaining stack to keep track of certain operations.
- It transitions between states and modifies the stack based on input symbol and top of the stack.
- Context-free languages often require remembering past symbols which stack allows.
- PDA accepts input string if after consuming entire string, it ends in accepting state and in some cases with stack in specific condition.

4) Example :

Consider language $L = \{a^n b^n \mid n \geq 1\}$, which consists of strings with equal number of a's followed by b's. This language is context-free and can be recognized by PDA as it can use the stack to "remember" the number of a's and then match these with equal number of b's.

Q.3 write and explain any 3 closure properties of CFL.

→ 1) Union

If L_1 and L_2 are context-free language, then their union $L_1 \cup L_2$ is also a context-free language

Union means combining all strings from both languages If a string belongs to either L_1 or L_2 is included in the union.

Example :-

If $L_1 = \{a^n b^n \mid n \geq 1\}$ and

$L_2 = \{a^m c^m \mid m \geq 1\}$ then the Union $L_1 \cup L_2$ is

$L_1 \cup L_2 = a^n b^n$ or $a^m c^m$: [both of which are context free]

2) Concatenation

If L_1 and L_2 are context-free languages, then the concatenation $L_1 L_2$ is also context free language.

Concatenation means combining strings from L_1 & L_2 such that first string comes from L_1 & then second comes from L_2.

Example :- If $L_1 = \{a^n b^n \mid n \geq 1\}$ and

$L_2 = \{b^m c^m \mid m \geq 1\}$ then the concatenation

$L_1 \cdot L_2 = \{a^n b^n b^m c^n \mid n, m \geq 1\}$,
which is also a Context-free language.

3) Kleene star
If L is Context-free language, then the Kleene star L^*, which consists of all strings that can be formed by concatenating zero or more strings from L, is also a Context-free language

Kleene star applies to language L, producing set of all possible strings that can be formed is concatenating any number of strings from L, including the empty string.

Example:
If $L = \{a^n b^n \mid n \geq 1\}$, then

$L^* = \{a^n b^n\}^*$

$= \{\epsilon, ab, aabb, \dots\}$.