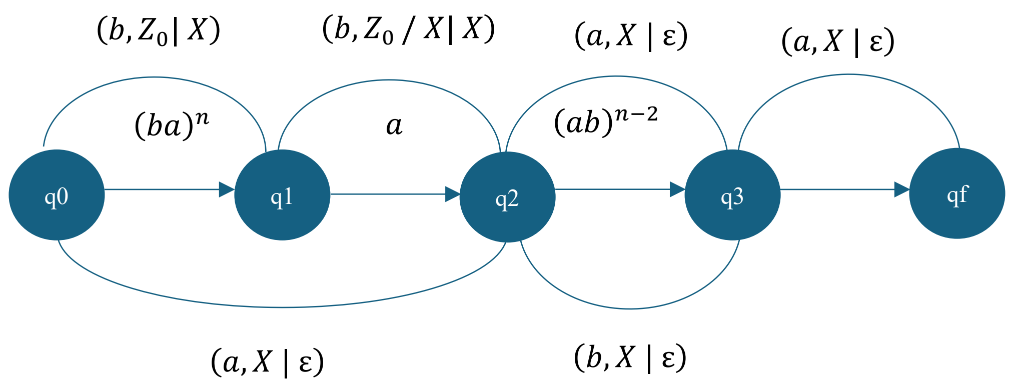
Question 1
Build a DPDA to show that the language $L = \{(ba)^n a(ab)^{n-2} \mid n > 2\}$ is deterministic context free.

$Q = \{q_0, q_1, q_2, q_3, q_f\}$
$\Sigma = \{a\ ,\ b\}$
$\Gamma = \{Z_0\ ,\ X\}$
$\delta: Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow Q \times \Gamma$

defined by:

Current State	**Input**	**Stack Top**	**Next State**	**Stack Operation**
q_0	b	Z_0	q_0	Push X
q_0	b	X	q_0	Push X
q_0	a	X	q_1	Pop X
q_1	a	X	q_2	Pop X
q_2	a	X	q_3	Pop X
q_3	a	X	q_3	Pop X
q_3	b	X	q_3	Pop X
q_3	ϵ	Z_0	q_f	-

Transitions:
(q0, b, Z0) -> (q0, XZ0)
(q0, b, X) -> (q0, XX)
(q0, a, X) -> (q1, ε)
(q1, a, X) -> (q2, ε)
(q2, a, X) -> (q3, ε)
(q3, a, X) -> (q3, ε)
(q3, b, X) -> (q3, ε)
(q3, ε, Z0) -> (qf, Z0)



Question 2
Prove that the language $L = \{banb^{2n}a^{n+1} \mid n > 0\}$
$L = \{(ba)^n a(ab)^{n-2} \mid n > 2\}$
over the alphabet ∑ = {a, b} is non-context free.
Use the pumping lemma with length.

Pumping Lemma for context-free languages

[1] Assume that the language $L = \{banb^{2n}a^{n+1} \mid n > 0\}$ is context-free.

[2] For any context-free language L
There exists a pumping length p such that any string $s \in L$ with $s \geq p$ can be decomposed into five parts $s = uvwxys$ satisfying the following conditions:

1. $Length(vwx)$ has at most ppp
2. vx is non-empty
3. For all $i \geq 0$, the string $uv^i wx^i y$ is also in L

[3] Chose suitable word s:
$s = ba^p b^{2p} a^{p+1}$
$Length(s) = 4p + 2$

[4] Five ways in which vwx can occur in the word:
1. Initial 'b' and some of the 'a's.
2. Some/all of the 'a's.
3. Some of the 'a's to 'b's and some of the 'b's.
4. Some or all of the 'b's.
5. The transition from 'b's to the final 'a's and some of the 'a's.

[5] show that pumping v and x results in a string that does not belong to L for each case

1. Suppose vwx consists of an **Initial 'b' and some of the 'a's**
- v and/or x would have 'b's and 'a's.
- pumping v and/or x, change the number of 'a's before the sequence of 'b's disrupting the pattern of s
- $Length(a)$ before 'b's must be p.

2. Suppose vwx consists of Some/all of the 'a's.
- v and x would have 'b's and 'a's.
- pumping v and x changes the number of 'a's disrupting the pattern of s
- $Length(a)$ before 'b's must be p.

3. Suppose vwx consists of 'a's to 'b's and some of the 'b's.
- v and x would have 'a's and 'b's.
- pumping v and/or x would disrupt the count of 'a's

4. Suppose vwx consists of Some or all of the 'b's
- v and x would have 'b's.
- pumping v and x changes the number of 'b's
- $Length(b)$ before 'b's must be $2p$.

5. Suppose vwx consists of 'b's to the final 'a's and some of the 'a's.
- v and x would have 'b's and 'a's.
- pumping v and/or x would disrupt the count of 'b's disrupting the pattern of s
- pumping v and x would disrupt the form of s
- $Length(b)$ before 'b's must be $2p$.

Thus, L is not a context-free language.

Question 3

Let L_1 be the grammar generating $(aa)*$.
Let L_2 be the grammar generating $(a+b)*ba(a+b)*$.

First provide the grammars generating L_1 and L_2 respectively.
Then apply the applicable theorem of Chapter 17 to determine L_1 L_2.

[1] Define grammar for L_1

$G_1 = (V_1, \Sigma_1\ P_1\ , S_1)$

Σ_1: **Terminal(s)**	a
V_1: **Non-terminal(s)**	S_1
P_1: **Production Rule(s)**	P_1 $S_1 \to aaS_1$ $S_1 \to \epsilon$

[2] Define grammar for L_2

$G_2 = (V_2, \Sigma_2\ P_2\ , S_2)$

Σ_2: **Terminal(s)**	a, b
V_2: **Non-terminal(s)**	$S_2\ , A\ , B$
P_2: **Production Rule(s)**	P_2 $S_1 \to AB$ $A \to aA \mid bA \mid \epsilon$ $b \to bAa \mid bBAa \mid ba$

[3] Theorem on concatenation of context-free languages

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

Σ: **Terminal(s)**	$\Sigma_1 \cup \Sigma_2$
V: **Non-terminal(s)**	$V_1 \cup V_2 \cup \{S\}$
P: **Production Rule(s)**	P P_1 $S_1 \to aaS_1$ $S_1 \to \epsilon$ P_2 $S_1 \to AB$ $A \to aA \mid bA \mid \epsilon$ $b \to bAa \mid bBAa \mid ba$ $S \to S_1S_2$

Thus

Σ: **Terminal(s)**	a, b
V: **Non-terminal(s)**	$S, S_1, S_2\ , A, B$
P: **Production Rule(s)**	P $S_1 \to aaS_1$ $S_1 \to \epsilon$ $S_1 \to AB$ $A \to aA \mid bA \mid \epsilon$ $b \to bAa \mid bBAa \mid ba$ $S \to S_1S_2$

Question 4
Decide whether the grammar given below generates any words
S → XY
X → SY
Y → SX
X → a
Y → b

[1] Derivation process
$S \to XY$

[2] Derivation process
$X \to SY$
$Y \to SX$

1. $S \to XY$
2. $X \to SY$
   $S \to (SY)Y \to SYY$
3. $Y \to SX$
   $SYY \to S(SX)Y \to SSXY$

[3] Derivation process
1. $S \to XY$
2. $X \to a$
   $S \to aY$
3. $Y \to b$
   $S \to ab$

$S \to XY \to aY \to ab$ can be generated.
Therefore at least one word can be generated.
Thus, the grammar does generate words.