1. Give a context-free grammar for each of the following languages.

a) L= {w | w contains even number of 0's}

= $S \longrightarrow 1S|0T|\varepsilon$

$T \longrightarrow 0S|1T$

b) L={w | w contains twice as many 1s as 0s}

= $S \longrightarrow SS|A|B|C$

$A \longrightarrow A011|0A11|01A1|011A|\varepsilon$

$B \longrightarrow B110|1B10|11B0|110B|\varepsilon$

$C \longrightarrow C101|1C01|10C1|101C|\varepsilon$

c) L={w | w contains even number of 0s and 1s}

= $S \longrightarrow 0X|1Y|\varepsilon$

$X \longrightarrow 0S|1Z$

$Y \longrightarrow 1S|0Z$

$Z \longrightarrow 0Y|1X$

d) L={w | where each 0's is followed by at least as many 1's}

= $S \longrightarrow AS|\varepsilon$

$A \longrightarrow 0A1|1A|\varepsilon$

e) L(G) = { $a^ib^jc^k$ | i, j, k ≥ 0 and i=j or i=k}. ∑={a,b,c}

= $S \longrightarrow AC|\ S'$
$A \longrightarrow aAb|\varepsilon$
$C \longrightarrow cC|\varepsilon$
$S' \longrightarrow aBc|B$
$B \longrightarrow bB|\varepsilon$

f)) L(G) = { $a^ib^jc^k$ | j > i+k}. ∑={a,b,c}

= $S \longrightarrow ABC$

$A \longrightarrow aAb|\ \varepsilon$

$B \longrightarrow bB|\ b$

$C \longrightarrow bCc|\ \varepsilon$

g) L(G) = { $a^n b^m$ | 0 < n < m < 3n}.∑={a,b}

= $S \longrightarrow aSbb \mid aSbbb \mid Zb$
$Z \longrightarrow aZb \mid ab$

h) L(G) = set of all strings w over {a, b} such that w is not palindrome.

= $Y \longrightarrow aYa \mid bYb \mid aZb \mid bZa$
$Z \longrightarrow aZ \mid bZ \mid \varepsilon$

i) L={w | w = w^R AND |w| is even, w is a palindrome}

= $S \longrightarrow A0A \mid B1B \mid \varepsilon$
$A \longrightarrow 1A \mid 0A \mid \varepsilon$
$B \longrightarrow 1B \mid 0B \mid \varepsilon$

j) L(G) = { $a^i b^j c^k$ | i, j, k ≥ 0 and i=j or j=k}. ∑={a,b,c}

= $S \longrightarrow AC \mid S'$
$A \longrightarrow aAb \mid C$
$C \longrightarrow cC \mid \varepsilon$
$S' \longrightarrow A'B$
$A' \longrightarrow aA' \mid \varepsilon$
$B \longrightarrow bBb \mid A'$

k) L(G) = { $a^n b^m c^m d^{2n}$ | n ≥ 0, m > 0}

= $S \longrightarrow aBdd \mid A$
$A \longrightarrow aSdd \mid \varepsilon$
$B \longrightarrow bBc \mid bc$

l) L= {w | w contains at least 4 a's}

= $S \longrightarrow RaRaRaRaR$
$R \longrightarrow bR \mid aR \mid \varepsilon$

2. What does the following CFGs do?

a) $S \longrightarrow ZSZ \mid 0$
$Z \longrightarrow 0 \mid 1$

= L={w | the length of w is odd and its middle is 0}

b) $S \longrightarrow 0E0 \mid 1E1 \mid \varepsilon$
$E \longrightarrow 1E \mid 0E \mid \varepsilon$

= L={w | w starts and ends with the same symbol}

c) $S \longrightarrow AB$

$A \longrightarrow 0A1 | \varepsilon$

$B \longrightarrow 1B | \varepsilon$

= $L(G) = \{0^m 1^{m+n} \mid n, m \geq 0\}$ over the terminals {0,1}

d) $S \longrightarrow \varepsilon | 1S1S1S0S \mid 1S1S0S1S \mid 1S0S1S1S \mid 0S1S1S1S$

= L={w | w contains thrice as many 1s as 0s}

e) $S \longrightarrow aSbb \mid aSb \mid \varepsilon$

= $L(G) = \{a^n b^m \mid 2n \geq m \geq n \geq 0\}$ over the terminals {0,1}

3. Convert the following Regular expressions to a CFG.

a) a(b|c*)

= $S \longrightarrow aX$

$X \longrightarrow b | C$

$C \longrightarrow Cc | \varepsilon$

b) 0*1(0 + 1)*

= $S \longrightarrow A1B$

$A \longrightarrow 0A | \varepsilon$

$B \longrightarrow 0B \mid 1B \mid \varepsilon$

c) (a + b)*(a* + (ba)*)

= $V \rightarrow WX$

$W \rightarrow aW$

$W \rightarrow bW$

$W \rightarrow \varepsilon$

$X \rightarrow Y$

$X \rightarrow Z$

$Y \rightarrow aY$

$Y \rightarrow \varepsilon$

$Z \rightarrow baZ$

Z ⟶ ε

d) (a+b)* aa (a+b)*

= S⟶ AaaA

A⟶ aA|bA| ε

e) a* + a(a|b)*

= S⟶ X|Y

X⟶ aX| ε

Y⟶ aZ

Z⟶ aZ|bZ| ε

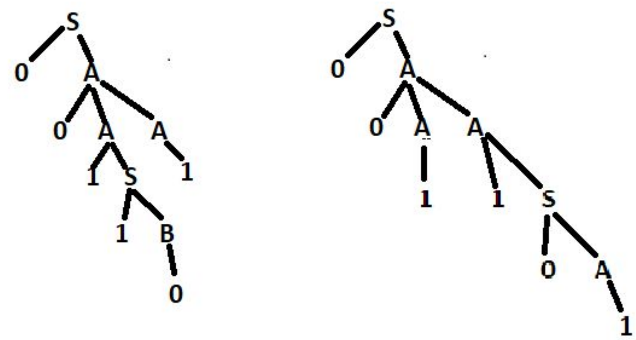
4. Consider the following context-free grammar ∑={0,1}. Give leftmost and rightmost derivations for the following strings and check parse-tree ambiguity.

a) S ⟶ 0A | 1B
A ⟶ 0AA| 1S | 1
B ⟶ 0S |1BB| 0

Strings: 001101

leftmost derivation:	**rightmost derivation:**
S ⟶ 0A ⟶00AA ⟶001A ⟶0011S ⟶00110A ⟶001101	S ⟶ 0A ⟶00AA ⟶00A1 ⟶001S1 ⟶0011B1 ⟶001101

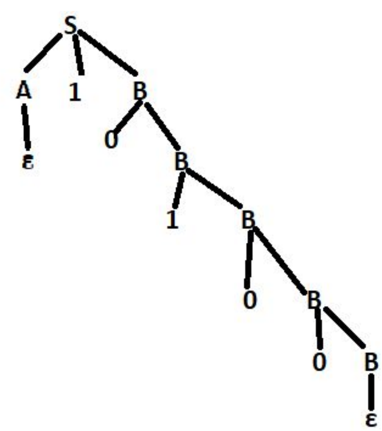
we can find two parse trees for this grammar, so the grammar is ambiguous.



b) S ⟶ A 1 B
A ⟶ 0A | ε
B ⟶ 0B |1B| ε

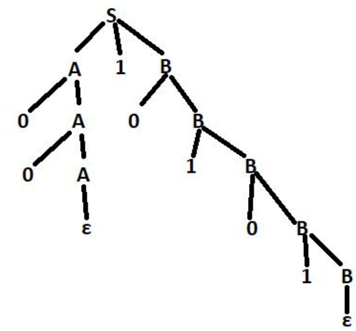
Strings: 10100, 0010101

= for string 10100:

leftmost derivation: **S ⟶ A1B** **⟶ ϵ1B** **⟶ 10B** **⟶ 101B** **⟶ 1010B** **⟶ 10100ϵ** **⟶ 10100**	**rightmost derivation:** **S⟶ A1B** **⟶ A10B** **⟶ A101B** **⟶ A1010B** **⟶ A10100** **⟶ ϵ10100** **⟶ 10100**	**Parse Tree:** 

for string 0010101:

leftmost derivation: **S⟶ A1B** **⟶ 0A1B** **⟶ 00A1B** **⟶ 00ε1B** **⟶ 001B** **⟶ 0010B**	**rightmost derivation:** **S⟶ A1B** **⟶ A10B** **⟶ A101B** **⟶ A101ε** **⟶ 0A101** **⟶ 00A101**	**Parse Tree:**

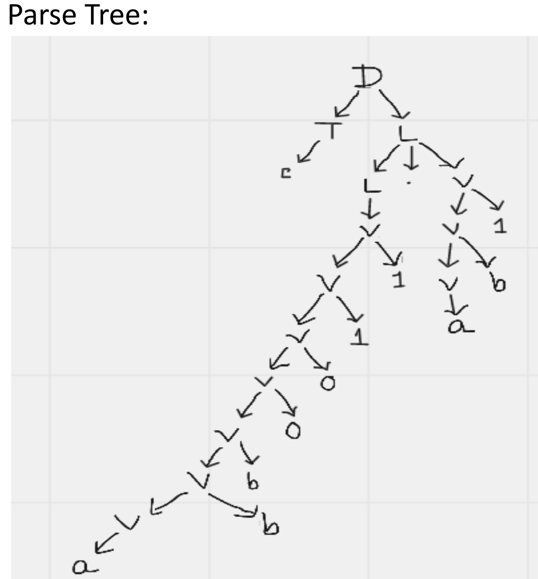
→ 00101B **→ 001010B** **→ 0010101B** **→ 0010101ε** **→ 0010101**	**→ 001A101** **→** **0010ε101** **→ 0010101**	

The grammar is unambiguous since only one parse tree is possible for every string.

c) D → TL
T → c | Tc
L → L.V | V
V → a | b | 0 | 1| Va | Vb | V0 | V1

Strings: cabb0011.ab1 (Rightmost derivation)

=

rightmost derivation	Parse Tree:
D→ TL **→ TL.V** **→ TL.V1** **→ TL.Vb1** **→ TL.ab1** **→ TV.ab1** **→ TV1.ab1** **→ TV11.ab1** **→ TV011.ab1** **→ TV0011.ab1** **→ TVb0011.ab1** **→ TVbb0011.ab1** **→ Tabb0011.ab1** **→ cabb0011.ab1**	

The grammar is unambiguous since only one parse tree is possible for every string.

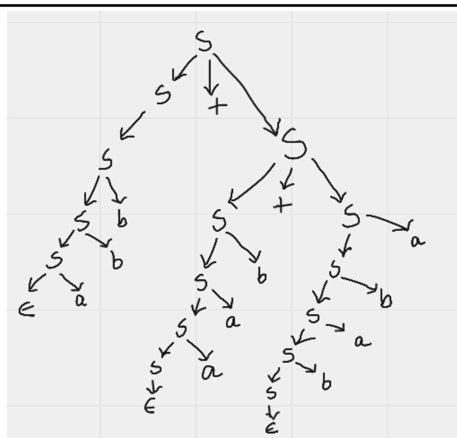
d) $S \rightarrow S + S$

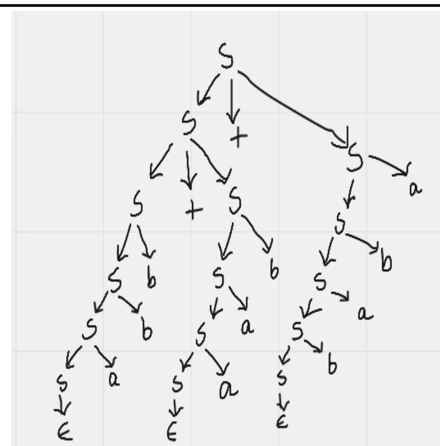
$S \rightarrow Sa \mid Sb \mid \varepsilon$

String: abb + aab + baba

Leftmost Derivation:	Rightmost Derivation:
S → S + S	S → S + S
→ Sb + S	→ S + Sa
→ Sbb + S	→ S + Sba
→ Sabb + S	→ S + Saba
→ abb + S + S	→ S + Sbaba
→ abb + Sb + S	→ S + baba
→ abb + Sab + S	→ S + S + baba
→ abb + Saab + S	→ S + Sb + baba
→ abb + aab + Sa	→ S + Sab + baba
→ abb + aab + Sba	→ S + Saab + baba
→ abb + aab + Saba	→ S + aab + baba
→ abb + aab + Sbaba	→ Sb + aab + baba
→ abb + aab + baba	→ Sbb + aab + baba
	→ Sabb + aab + baba
	→ abb + aab + baba

we can find two parse trees for this grammar, so the grammar is ambiguous.

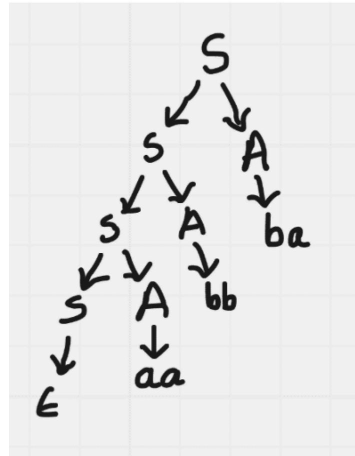




e) $S \rightarrow SA \mid \varepsilon$

$A \rightarrow aa \mid ab \mid ba \mid bb$

String: aabbba

Leftmost Derivation:	Rightmost Derivation:	
S ⟶ SA ⟶SAA ⟶SAAA ⟶aaAA ⟶aabbA ⟶aabbba	S ⟶ SA ⟶Sba ⟶SAba ⟶Sbbba ⟶SAbbba ⟶Saabbba ⟶aabbba	

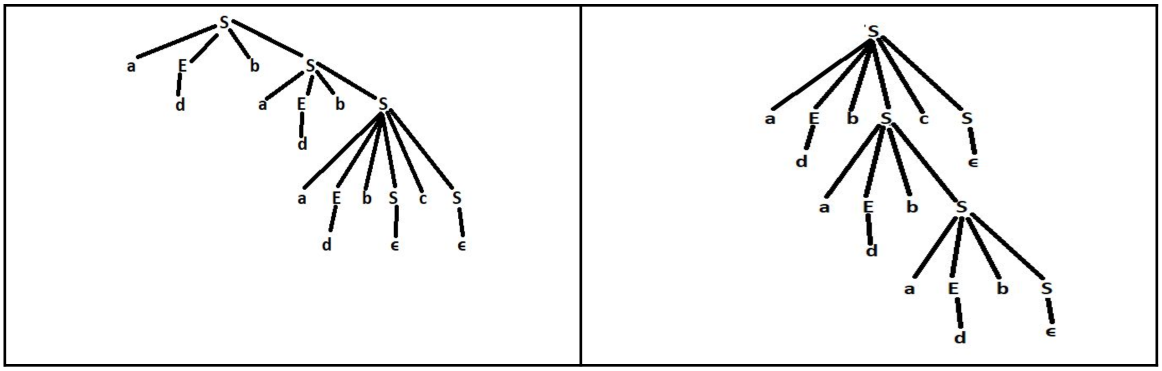
The grammar is unambiguous since only one parse tree is possible for every string.

f) S ⟶ aEbS
S ⟶ aEbScS | ϵ
E ⟶ d

String: adbadbadbc

Leftmost Derivation:	Rightmost Derivation:
S ⟶ aEbS ⟶adbS ⟶adbaEbS ⟶adbadbS ⟶adbadbaEbScS ⟶adbadbadbScS ⟶adbadbadbc	S ⟶ aEbS ⟶aEbaEbScS ⟶aEbaEbScϵ ⟶ aEbaEbSc ⟶ aEbaEbaEbSc ⟶ aEbaEbaEbϵc ⟶ aEbadbadbc ⟶ adbadbadbc

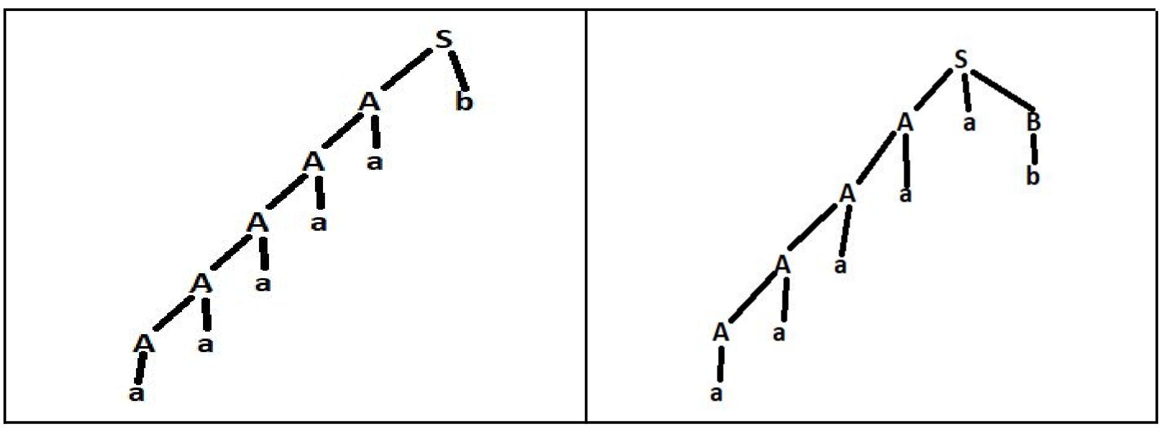
we can find two parse trees for this grammar, so the grammar is ambiguous.



5. Are the following CFGs ambiguous? Are they inherently ambiguous? If not, then give its unambiguous representation.

a) $S \rightarrow Ab \mid AaB$
$A \rightarrow a \mid Aa$
$B \rightarrow b$

= For the string aaaaab, we can find two parse trees. So the grammar is ambiguous.

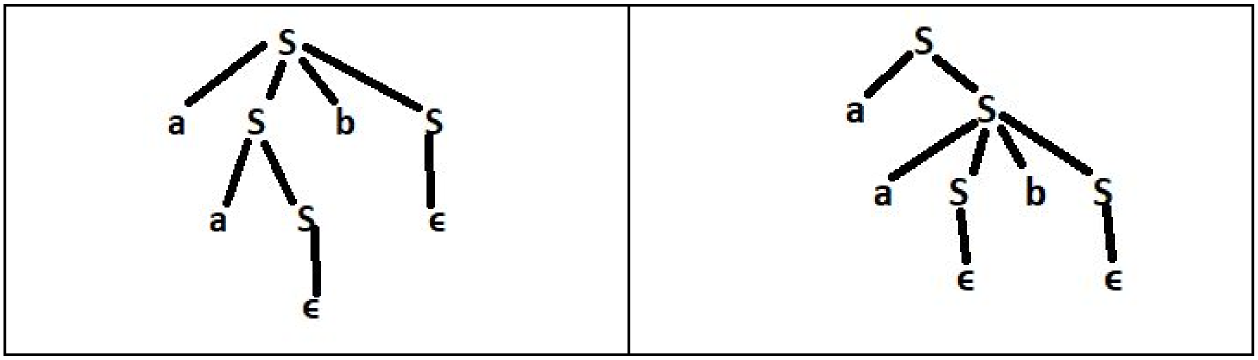


The grammar is ambiguous but not inherently ambiguous. So, equivalent unambiguous CFG:

$S \rightarrow A'b|ab$
$A' \rightarrow A'a|aa$

b) $S \rightarrow aS \mid aSbS \mid \epsilon$

= For the string aab, we can find two parse trees. So the grammar is ambiguous.



The grammar is ambiguous but not inherently ambiguous. So, equivalent unambiguous CFG:

$S \longrightarrow aS \mid aTbS \mid \epsilon$
$T \longrightarrow aTbT \mid \epsilon$