Q1: A: let the G the grammar below:

\[ S \rightarrow aAcbE \]
\[ A \rightarrow Ab / b \]
\[ B \rightarrow d \]

Parse the word "abcede" accept or not accept by using the bottom-up method?

B: Find the RE that define the same language accept by follow TG by using Klean's Theorem?

Q2 A: How to represent Turing machine for this regular expression

\[(a+b) b(a+b)^* \]

B: Convert the mealy machine into a Moore machine?

Q3: A: Draw the finite automata from these production rules

1. \[ S \rightarrow aaS / bbS / abX / baX / ^ \]
   \[ X \rightarrow aaX / bbX / abS / baS \]

2. \[ S \rightarrow aA / bB \]
   \[ A \rightarrow aS / a \]
   \[ B \rightarrow bS / b \]

3. \[ S \rightarrow aM / bS / ^ \]
   \[ M \rightarrow bS / aF / ^ \]
   \[ F \rightarrow aF / bF / ^ \]

4. \[ S \rightarrow aM / bS / ^ \]
   \[ M \rightarrow bS / aF / ^ \]
   \[ F \rightarrow aF / bF \]
B: Define the following:
1- Terminal symbol
2- None terminal symbol

Q4: A: let the G = (S); { ( , ) },P,S)
Where P denoted as:
S → ( ) / (S) / SS
Is the string:
• ((( ( ) ( )) )) in L(G)?
• Prove the grammar G ambiguous or not?

B: Write the language of type palindrome that is word for this equation
S X reverse (S) where S in any string in (a+b)*
And draw the Flowchart show how the word store in stack?

Q5: A: what the equivalence of moor and mealy machines?
B: apply the kleans theorem?

Q6: A: convert the CFG into a CNF:

S → aSa / bSb / Xa
X → ^ / b

B: How many start state for these following:
1- FA 2- TG 3- NFA

GOOD LUCK