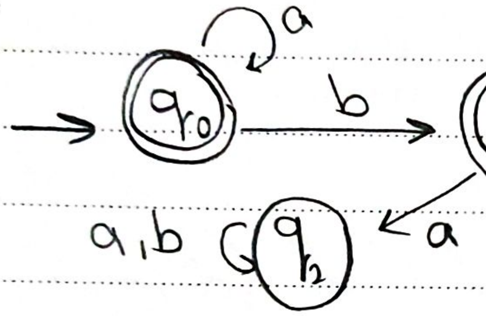
Theory sec "2"   / /

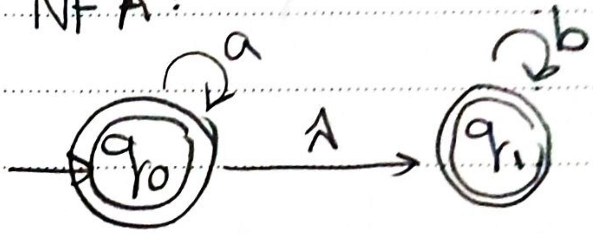
* Draw NFA and DFA of the following languages:

① $L = \{ a^n b^m ;\ n, m \geq 0 \}$

DFA:

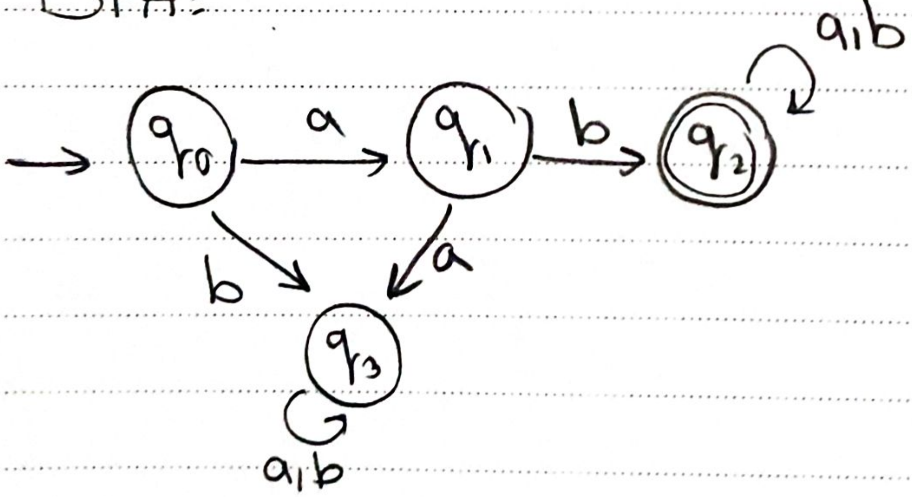


NFA:

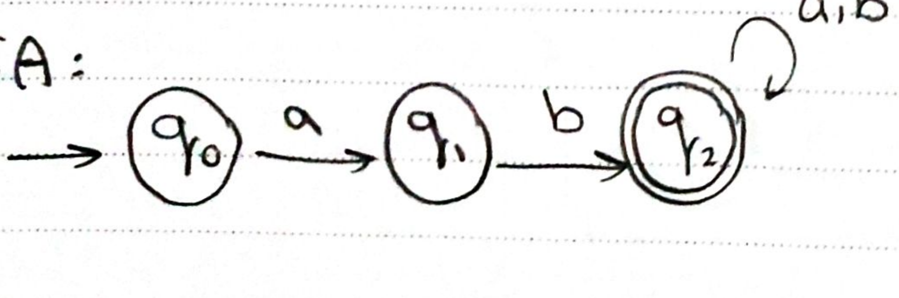


② L Σ {a,b} with prefix "ab"

DFA:

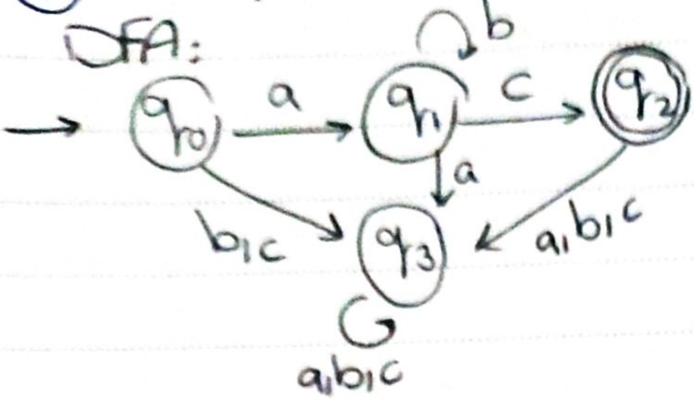


NFA:



③ $L = \{ a b^n c \; ; \; n \geq 0 \}$

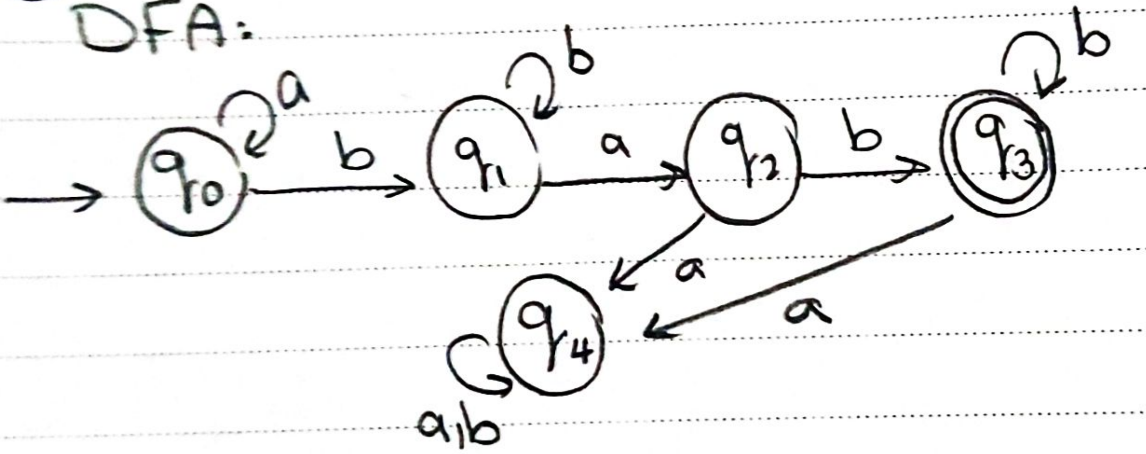
DFA:



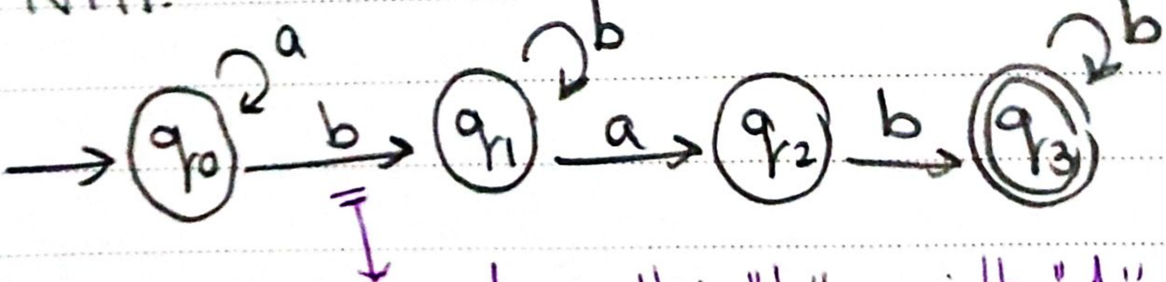
NFA:

④ $L = \{ a^n b^m a b b^j \; ; \; m \geq 1 \; ; \; n, j \geq 0 \}$

DFA:



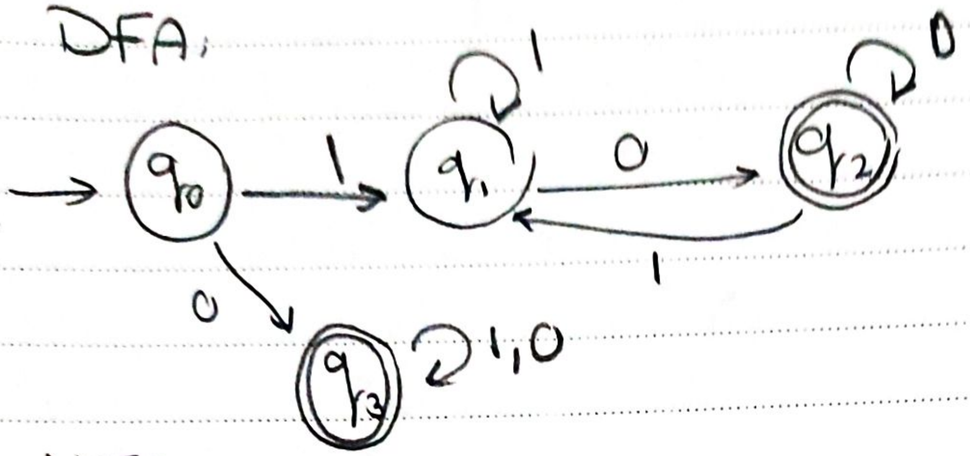
NFA:



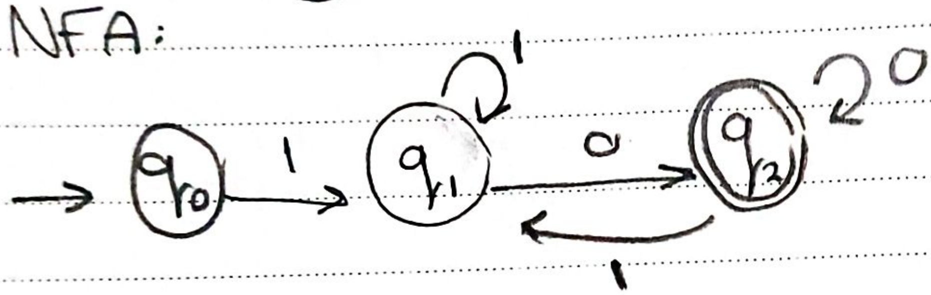
Can i replace this "b" with "λ", why?

5) L $\Sigma = \{0, 1\}$ accepts those string which starts with 1 and ends with 0.

DFA:



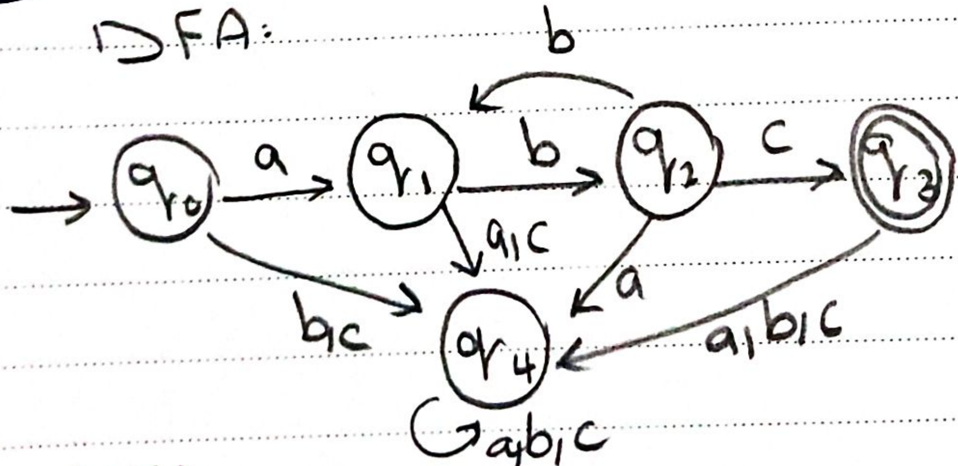
NFA:



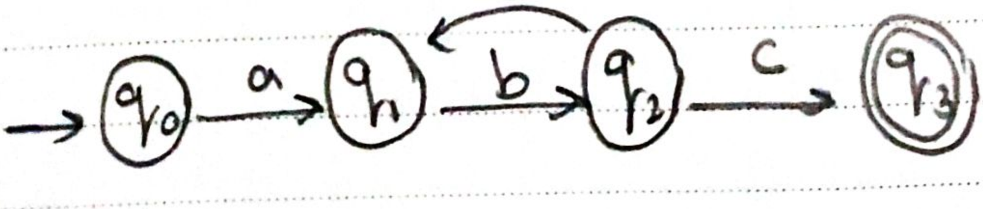
6) $L = \{a b^{2n+1} c \; ; n \geq 0\}$

n	
0	abc
1	abbbc
2	abbbbbc

DFA:

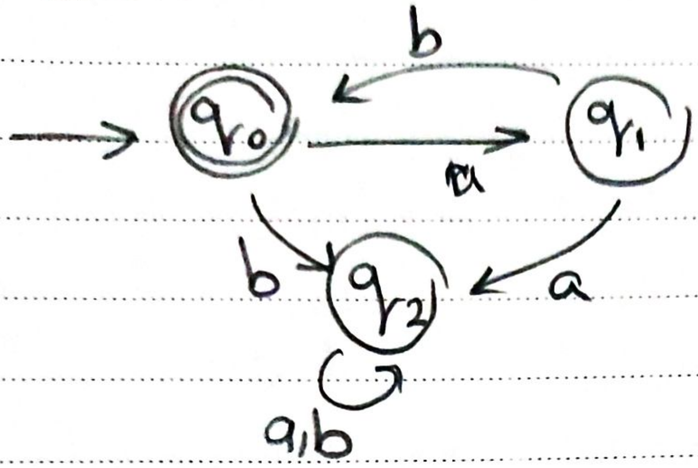


NFA:

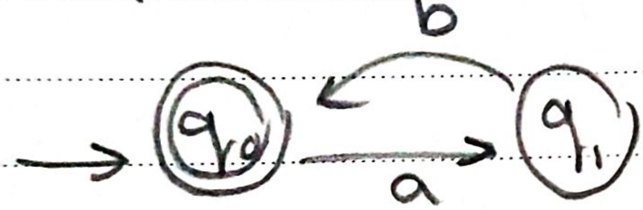


⑦ $L = \{ (ab)^n : n \geq 0 \}$

DFA:



NFA:



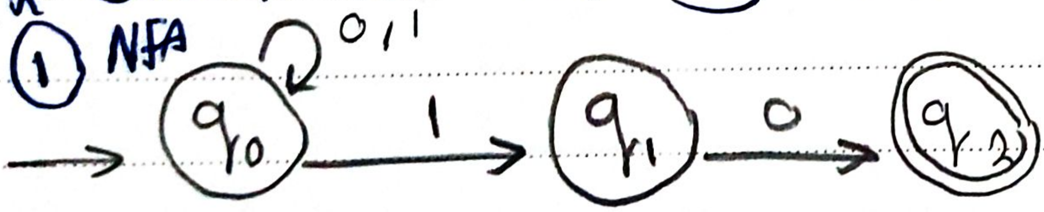
Convert from NFA to DFA:

→ Steps:

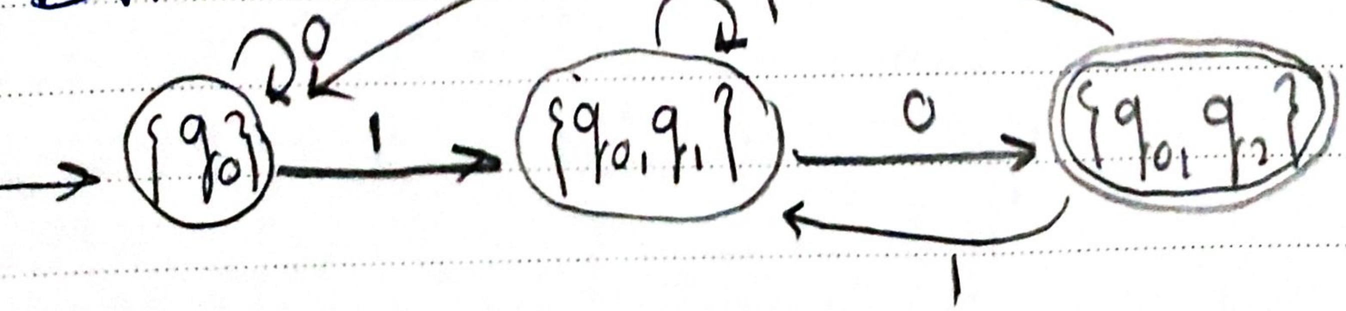
① Create a new DFA with start state of NFA
② for each state in DFA, find the set of states it can reach on a given input symbol.
③ if set of states obtained in step 2 is not already a state in the DFA, Add it.
④ Repeat step 2 and 3 untile no more new states added to DFA.
⑤ final state in the DFA is an accepting state if it contains at least one accepting state from NFA.

* Convert the following NFA to DFA:

① NFA

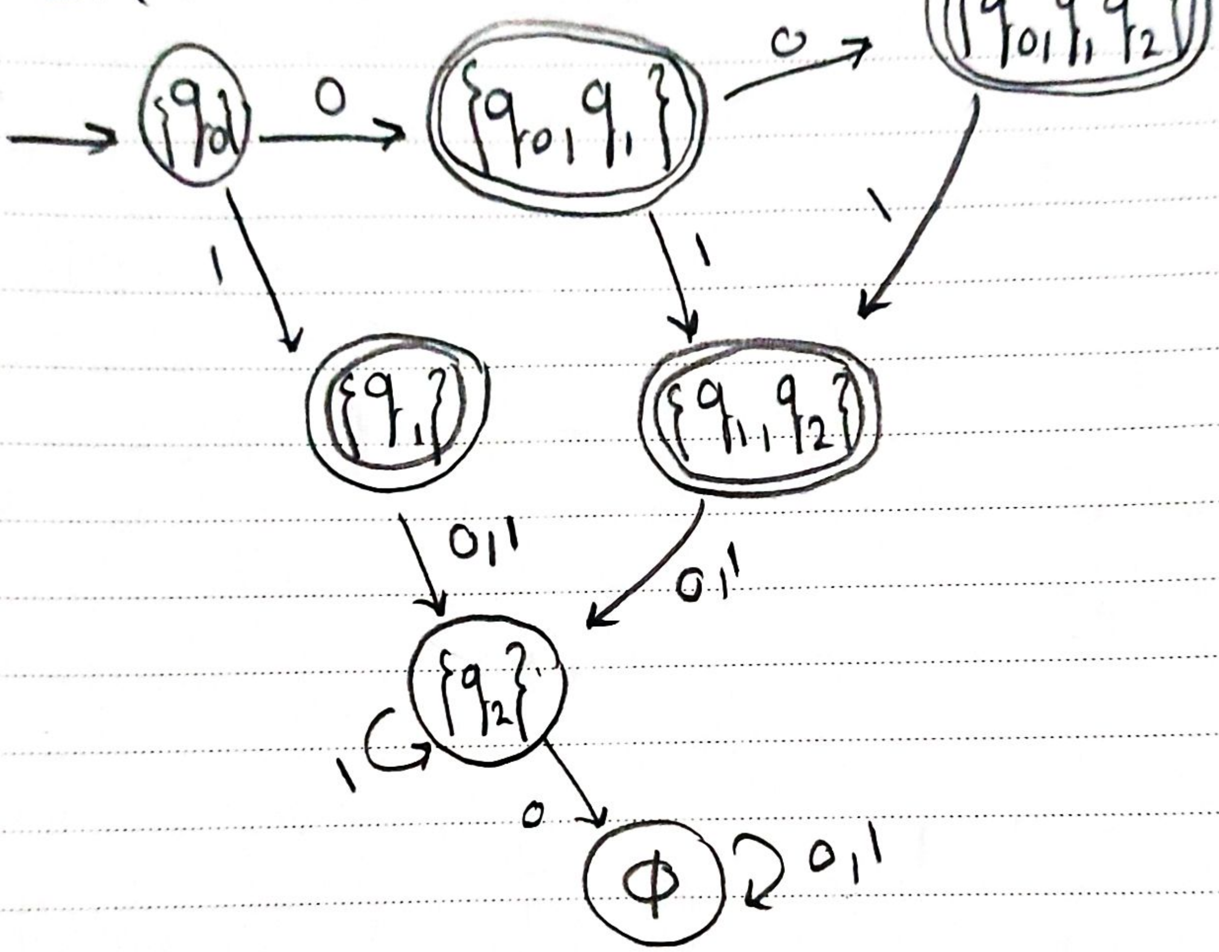


DFA:



② NFA:

DFA:



$\rightarrow$ move$(q_0, 0) = q_0, q_1$, move$(q_1, 0) = q_2$

$\rightarrow$ move$(q_0, 1) = q_1$, move$(q_1, 1) = q_2$

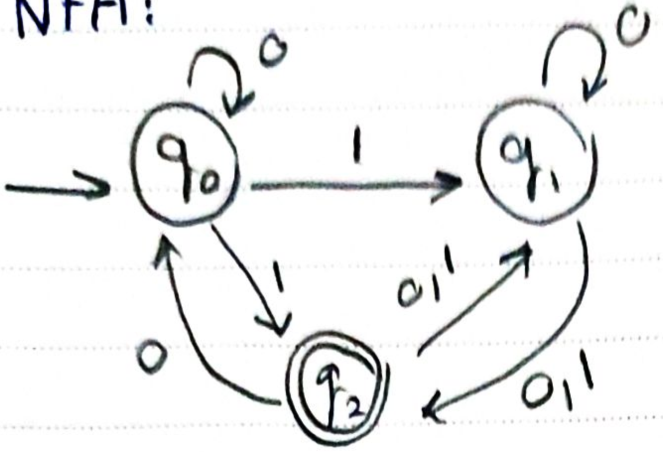
$\rightarrow$ move$(q_1, 0) = q_2$, move$(q_2, 0) = \phi$

$\rightarrow$ move$(q_1, 1) = q_2$, move$(q_2, 1) = q_2$

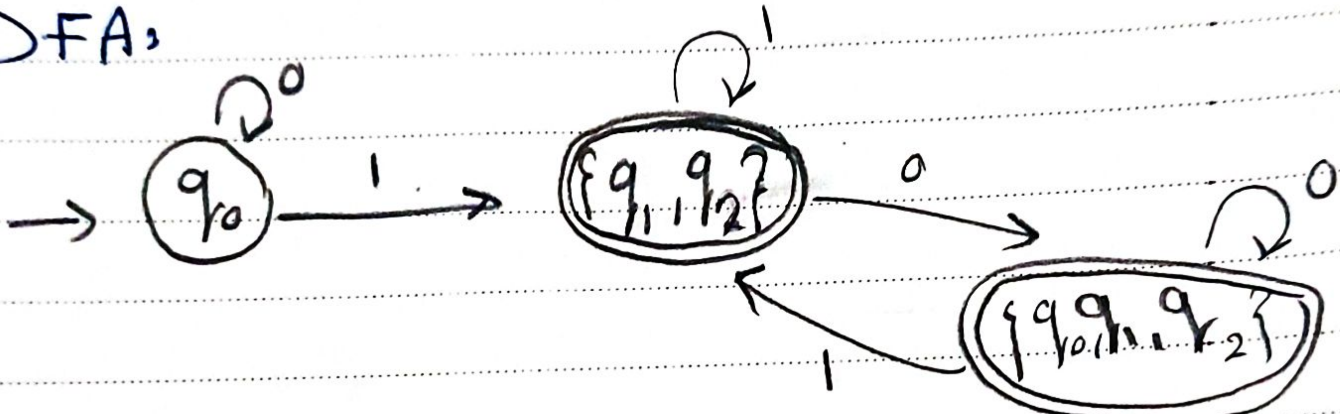
$\rightarrow$ move$(q_0, 0) = q_0, q_1$, move$(q_1, 0) = q_2$, move$(q_2, 0) = \phi$

$\rightarrow$ move$(q_0, 1) = q_1$, move$(q_1, 1) = q_2$, move$(q_2, 1) = q_2$

③ NFA:



DFA:



$move(q_1, 0) = q_1, q_2$, $move(q_2, 0) = q_0$

$move(q_1, 1) = q_2$, $move(q_2, 1) = q_1$

$move(q_0, 0) = q_0$, $move(q_1, 0) = q_1, q_2$, $move(q_2, 0) = q_1, q_2$

$move(q_0, 1) = q_1, q_2$, $move(q_1, 1) = q_2$, $move(q_2, 1) = q_1$