Example 1: This is a TEXfile.

$$(f * g) = \int_{-\infty}^{\infty} f\tau g(t - \tau)\, d\tau$$

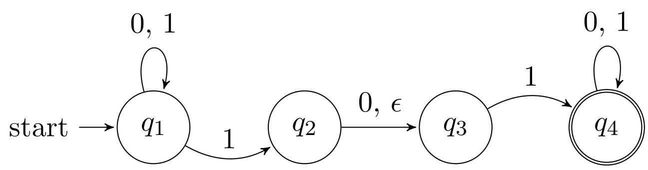
Example 2: *As NFA defined on Page 54 of ITC* textbook.

$Q = \{q_1, q_2, q_3, q_4\}$
$\sum = \{0, 1\}$
$F = \{q_4\}$
$q_0 = q_1$
$\delta = \{((q_1, 0), \{q_1\}), ((q_1, 1), \{q_1, q_2\}), ((q_1, \epsilon), \phi),$
$((q_2, 0), \{q_3\}), ((q_2, 1), \phi), ((q_2, \epsilon), \{q_3\}),$
$((q_3, 0), \phi), ((q_3, 1), \{q_4\}), ((q_3, \epsilon), \phi),$
$((q_4, 0), \{q_4\}), ((q_4, 1), \{q_4\}), ((q_4, \epsilon), \phi)\}$

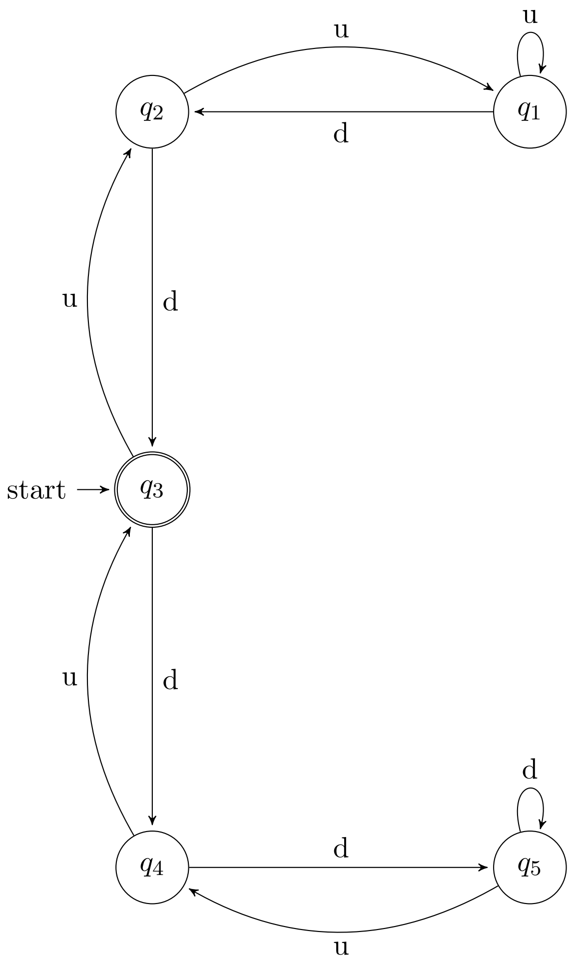
Transition Function in Table form:

	0	1	ϵ
q_1	$\{q_1\}$	$\{q_1,\ q_2\}$	ϕ
q_2	$\{q_3\}$	ϕ	$\{q_3\}$
q_3	ϕ	$\{q_4\}$	ϕ
q_4	$\{q_4\}$	$\{q_4\}$	ϕ

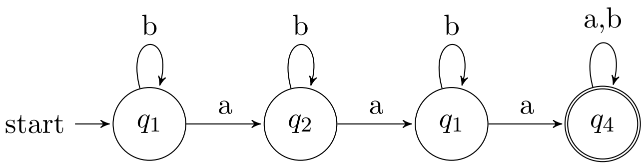
NFA in pictorial form:



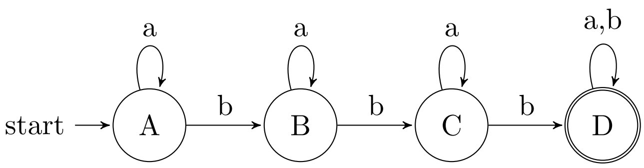
Example 3: ***DFA, state diagram of machine M***



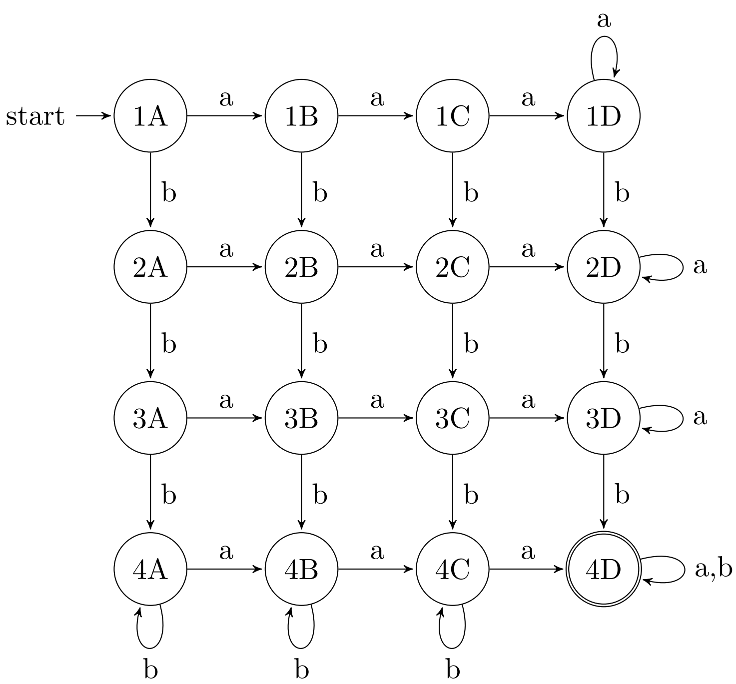
Example 4: **Machine DFA has 02 languages and it combine:**
{w| w has at least three a's}



{w| has at least three b's}

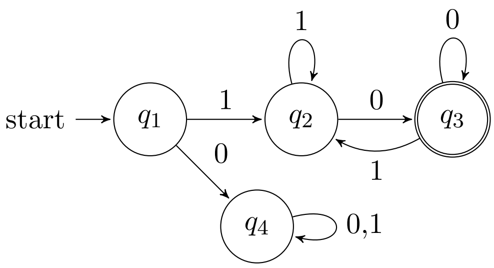


Combining them using the intersection construction for DFA machine:



Regular expression and it diagram DFA: $\mathbf{1}\sum^{*}\mathbf{0}$

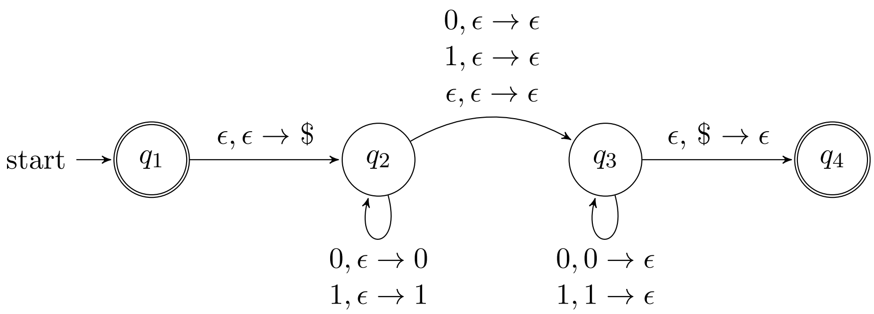
<u>**{w|w begin with a 1 and end with a 0}**</u>



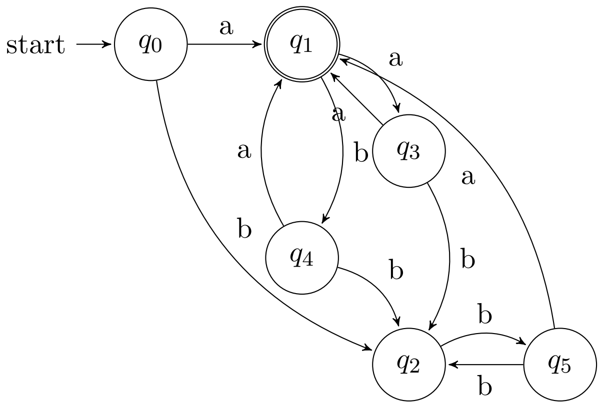
{w | w = w^R**,. that is, w is a palindrome}**

$$S \rightarrow 0S0 \mid 1S1 \mid 0 \mid 1 \mid \epsilon$$

Informal description: We begin by pushing the symbols read onto the stack. At each point we will nondeterministically guess if the middle of the string has been reached or if the next symbol read is the middle of the string and will not be put on the stack. Then we pop off the symbols from the stack if they match the input symbol read. If the symbol popped are exactly the same symbols that were pushed on earlier and the stack empties as the input is finished, ten accept. Otherwise, reject.

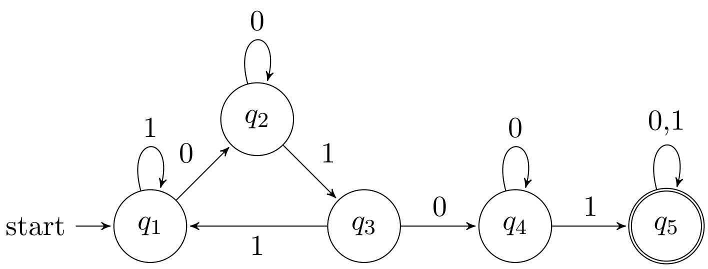


Example 4: D={w|w contains an even number of a's and odd number of b's and does not contain the substring ab}

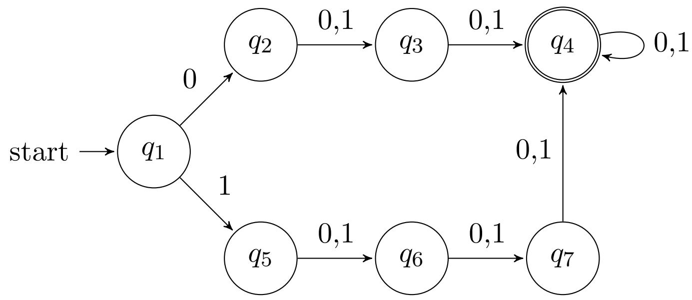


regular expression: $(aa|ba|bb)^*(a|b)$

4b. {w| w contains the substring 0101, e.x. w = x0101y for some x and y}

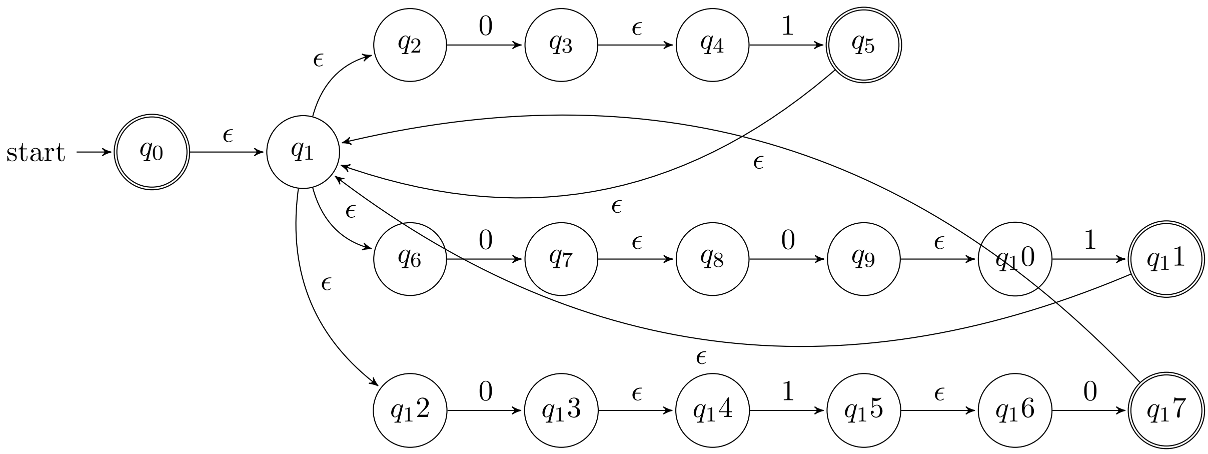


4c. { w| w starts with 0 and has odd length, or starts with 1 and has even length}



4c. $(0 \cup 1\sum)(\sum\sum)^*$ in regular expression

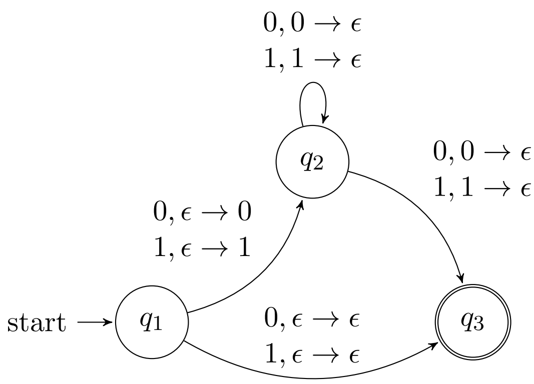
Example 5. Give an NFA recognizing the language $(01 \cup 001 \cup 010)^*$



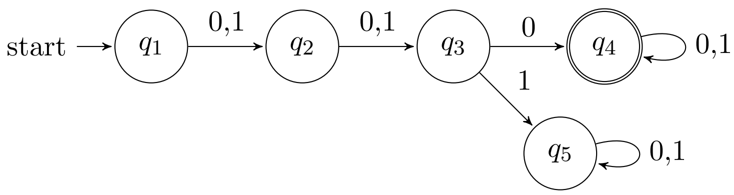
5a. {w | w starts and ends with the same symbol}

$$S \rightarrow 0T0 \mid 1T1 \mid 0 \mid 1$$
$$T \rightarrow 0T \mid 1T \mid \epsilon$$

Informal description: We will nondeterministically guess if the string has only one symbol in which case we accept it without using the stack. Then we will read every other symbol and nondeterministically guess if that is the last symbol read. If the last symbol read then matches the symbol on the stack and there is no more input we accept. Otherwise we reject.



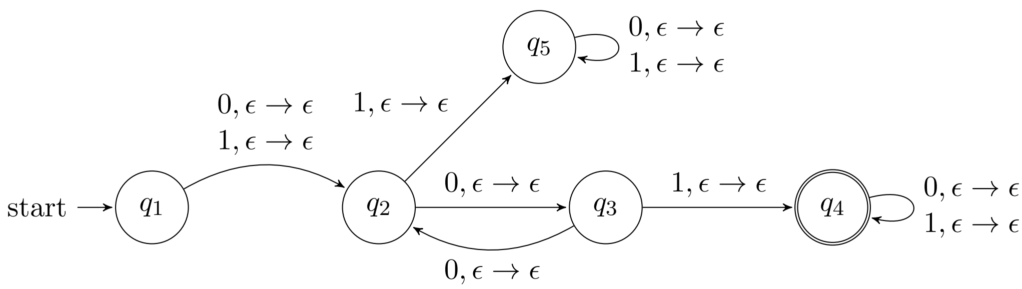
5b. { w| w has length at least 3 and third symbol is a 0}



5c. {w | the length of w is odd and its middle symbol is 0's}

S $\to$ 0 | 0S0 | 0S1 | 1S0 | 1S1

Formal description: The PDA scans across the string and push the symbol onto the stack. At some point it nondeterministically guesses where the middle is. It looks at the middle symbol. If that symbol is a 1, it rejects. Otherwise, it scans the rest of the string, and for each character scanned, it pops one element off of its stack. If the stack is empty when it finishes reading the input (ex: it correctly guessed the middle then it accepts.

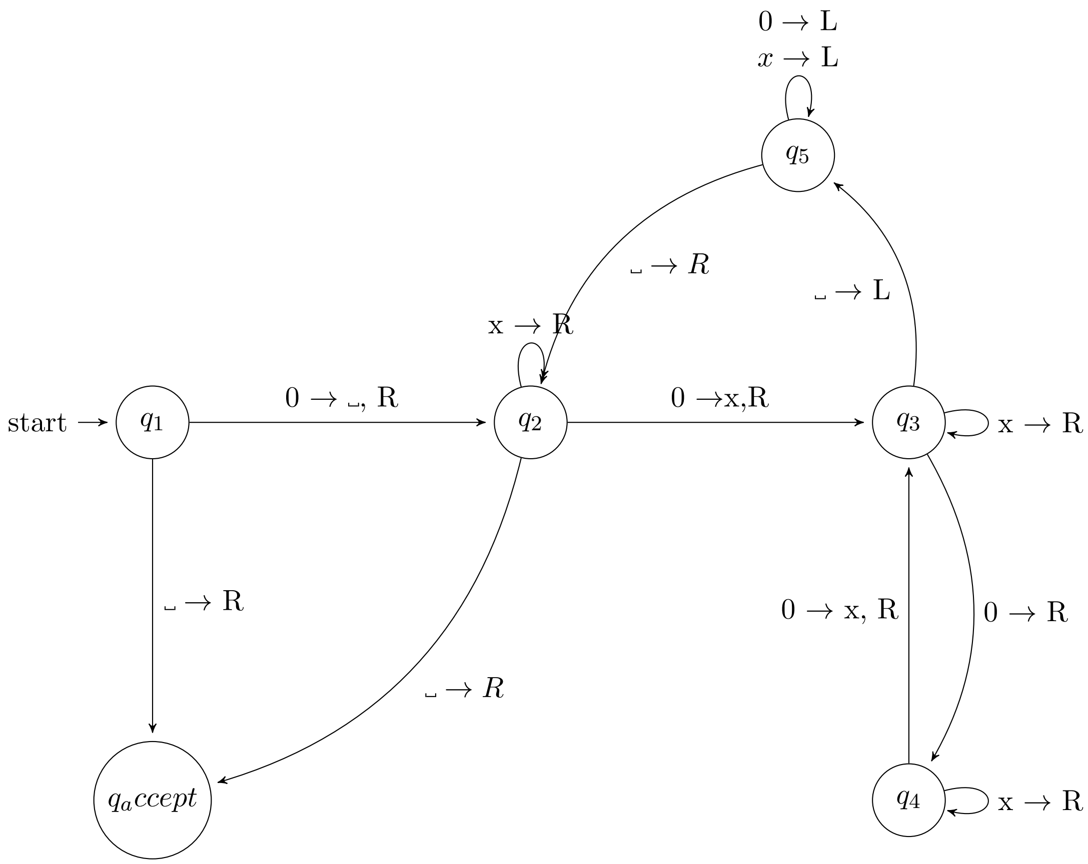


Example 6.Modify machine M2 to recognize even number of 0s and draw the state diagram.

$\sum$ ={0}

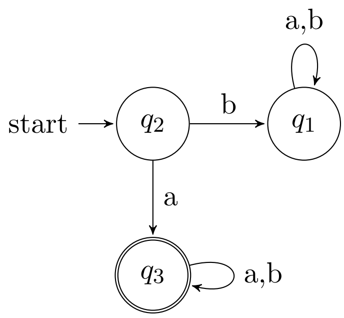
Γ={0,x,␣}

δ transition function as TM state diagram:

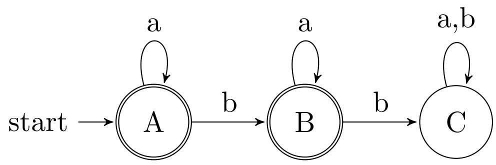


b. The folowing are DFA for two languages:

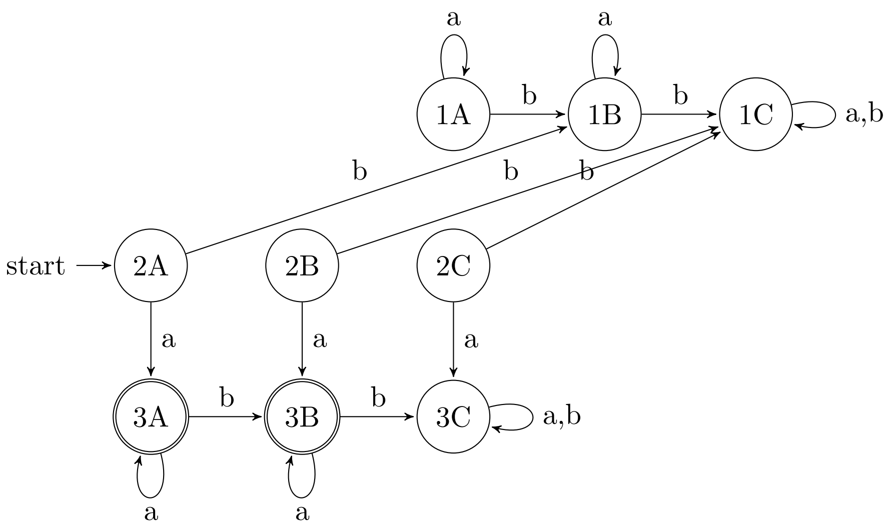
{w| w starts with and a}



{w| w has at most one b}

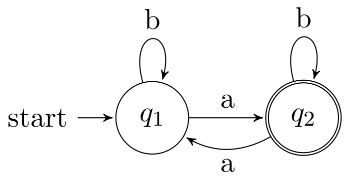


Combining them using the intersection construction gives the folowing DFA:

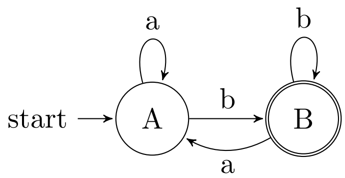


c. The folowing are DFA for two languages:

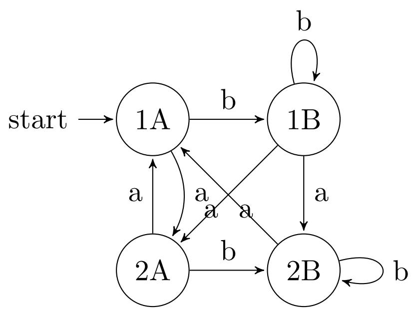
{w| w has an odd number of a's}



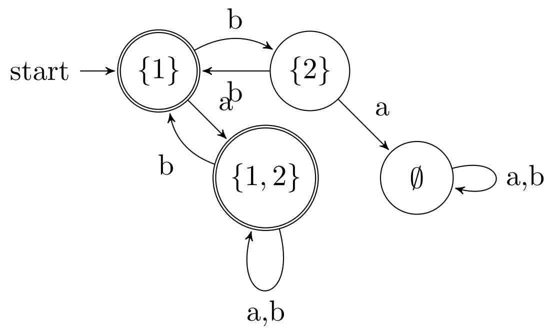
{w| w ends with a b}



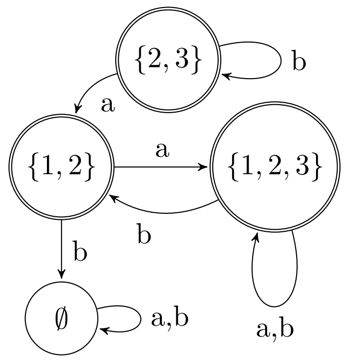
Combining them using the intersection construction gives the folowing DFA:



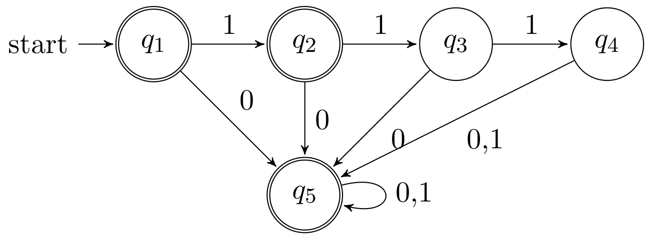
7. Use theorem 1.39 to convert the folowing to nondeterministic finite automata to equivalent deterministic finite automata.
a.



b.

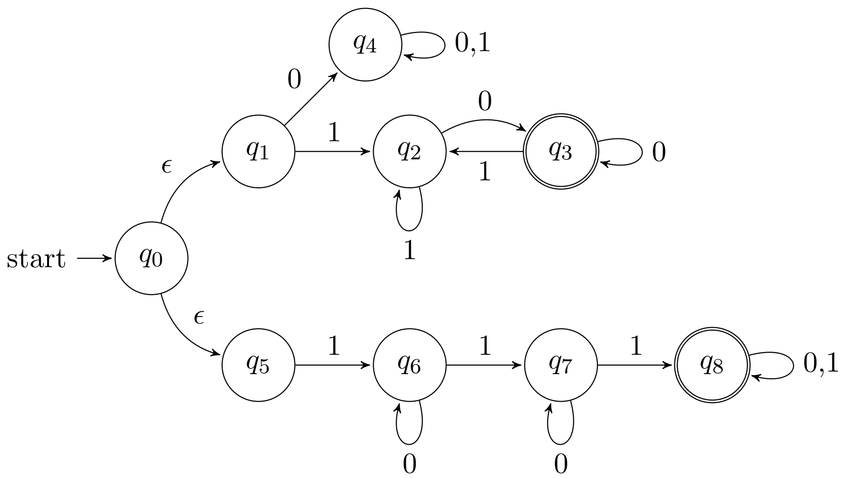


c. { w| w is any string except 11 and 111}

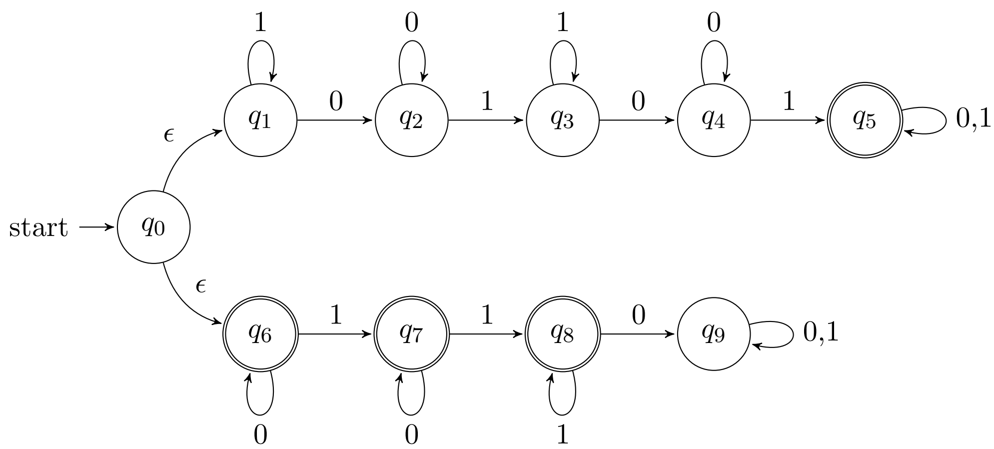


8. Use the construction in the proof of Theorem 1.45 to give the state diagrams of DFA recognizing the union of the languages described in.

a. Exercise 1.6a and 1.6b



b. Exercise 1.6 c and 1.6f



c. The empty state