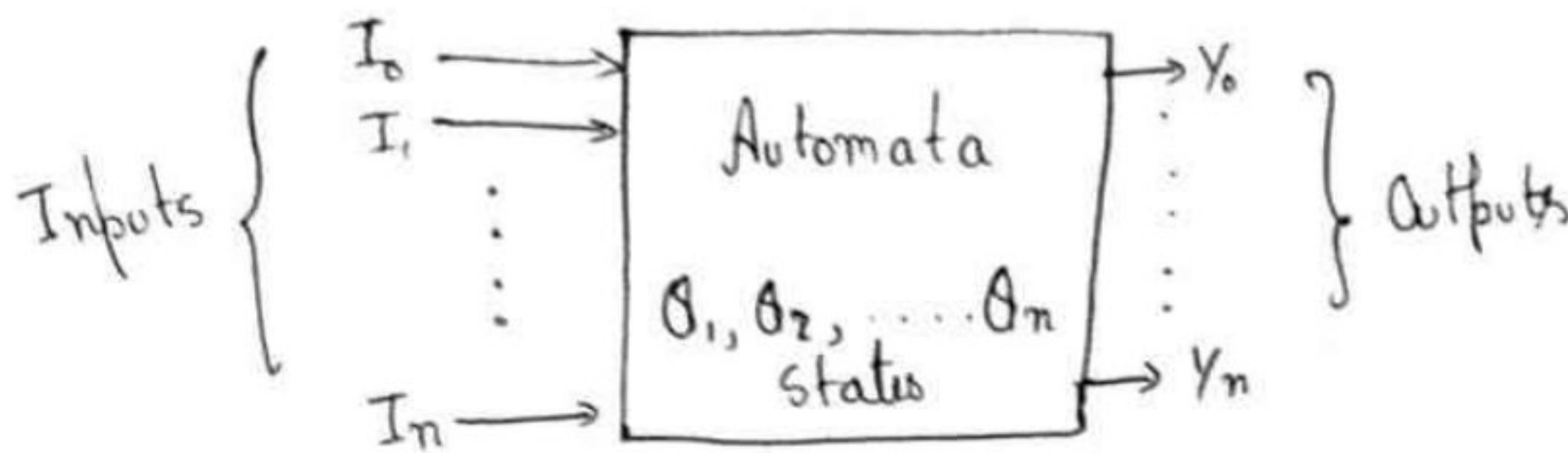


Automata Theory



A system should do / perform tasks without involvement of Human . This is called Automata

- i) Automata with Memory :- Output depends only on inputs
- ii) Moore Model:- Output depends only on states
- iii) Mealey Model :- Output depends only on states & inputs

- i) Symbols :- Basic building blocks (Letters + digits + characters)
Symbol
- ii) Alphabets (Σ) :- $\Sigma = \{0, 1\}$ $\Sigma = \{a, b\}$ $\Sigma = \{A, \dots, Z\}$
Always finite values
- iii) Strings (w) :- Sequence of characters @ symbols
 $|w| \rightarrow$ Finite String $\epsilon \rightarrow$ String does not have any value (NULL)
- iv) Languages:-
 - Represented using different alphabets
 - It may be finite @ infinite.

$L = \{a\}$
↓
 $\text{Infinite Language} = \{a, aa, aaa, \dots\}$

Finite Automata (DFA)

Deterministic Finite Automata

It can be represented by 5 Tuples

$$M = \{q, \Sigma, \delta, q_0, F\}$$

$q \rightarrow$ Finite Set of States

$\Sigma \rightarrow$ Finite Set of Alphabets.

$\delta \rightarrow$ Transition function ($\delta : q_0 \times \Sigma \rightarrow q_f$)

$q_0 \rightarrow$ Initial State

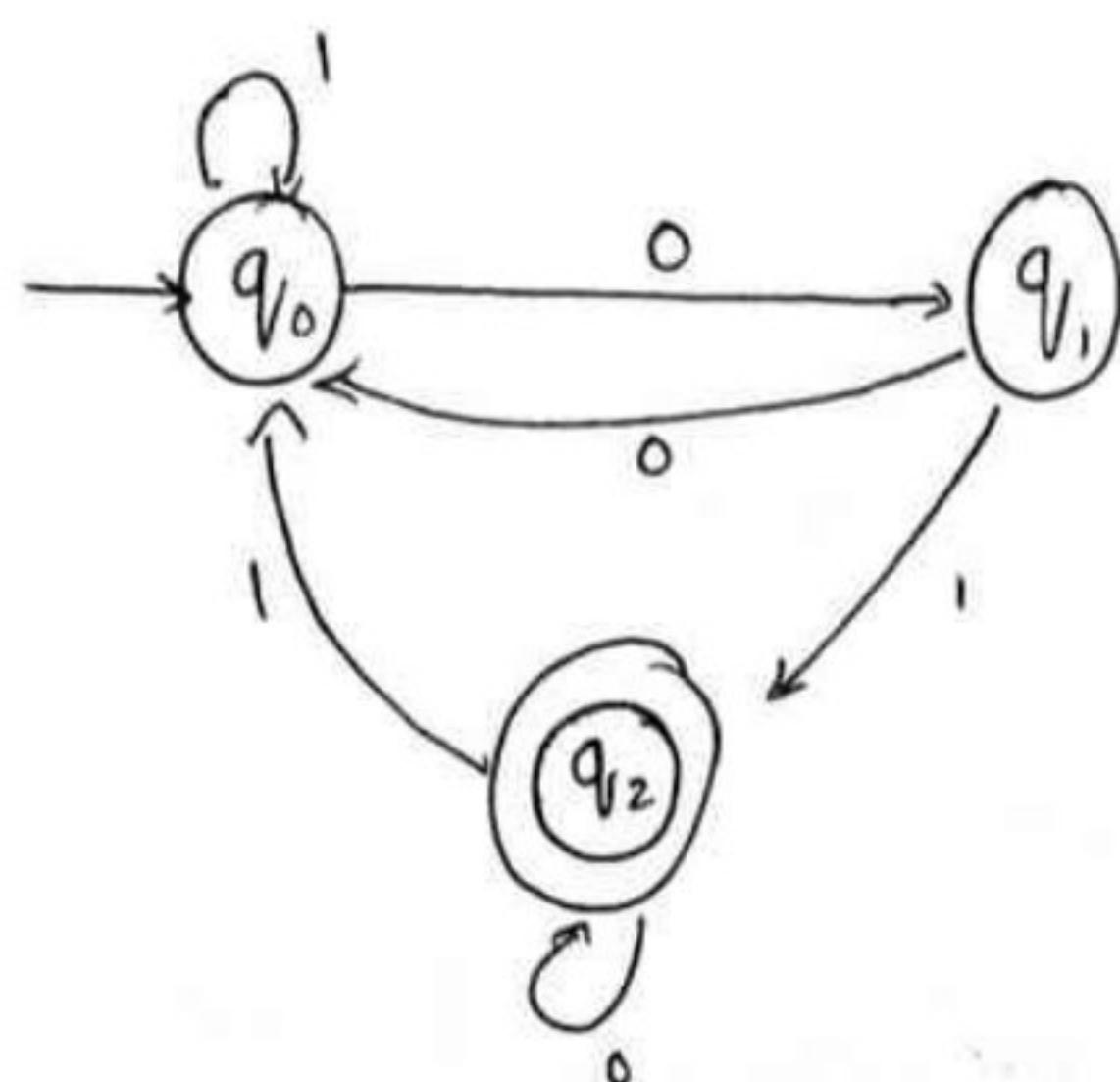
$F \rightarrow$ Final State ($F \subset q$)

→ Transition function (δ) :- Taking different inputs and producing different outputs.

→ Transition Table :- A Tabular format of different transitions

→ Transition Diagram :- Diagrammatic representation of different transitions.

\Rightarrow Representation of Transition State and Transition Diagram



- * Initial state has arrow and final state has double circle

P. S	I/P	N.S
q_{V_0}	0	q_{V_1}
q_{V_0}	1	q_{V_0}
q_{V_1}	0	q_{V_0}
q_{V_1}	1	q_{V_2}
q_{V_2}	0	q_{V_2}
q_{V_2}	1	q_{V_0}

P. S	N.S
0	1
q_{V_0}	q_{V_1}
q_{V_1}	q_{V_0}
q_{V_2}	q_{V_2}

$$M = \{ \{q_{V_0}, q_{V_1}, q_{V_2}\}, \{0, 1\}, \begin{matrix} A \\ \delta(q_{V_0}, 0) \rightarrow q_{V_1} \\ \delta(q_{V_0}, 1) \rightarrow q_{V_2} \\ \delta(q_{V_1}, 0) \rightarrow q_{V_0} \\ \delta(q_{V_1}, 1) \rightarrow q_{V_2} \\ \delta(q_{V_2}, 0) \rightarrow q_{V_1} \\ \delta(q_{V_2}, 1) \rightarrow q_{V_0} \end{matrix}, \begin{matrix} \epsilon \\ \delta(q_{V_0}, 0) \rightarrow q_{V_1} \\ \delta(q_{V_1}, 0) \rightarrow q_{V_2} \\ \delta(q_{V_2}, 0) \rightarrow q_{V_1} \end{matrix}, \begin{matrix} F \\ q_{V_2} \end{matrix} \}$$

=> Theory of Computation is divided as:-

① Automata Theory

- * Stores 1 bit data
- * Low Processing Speed

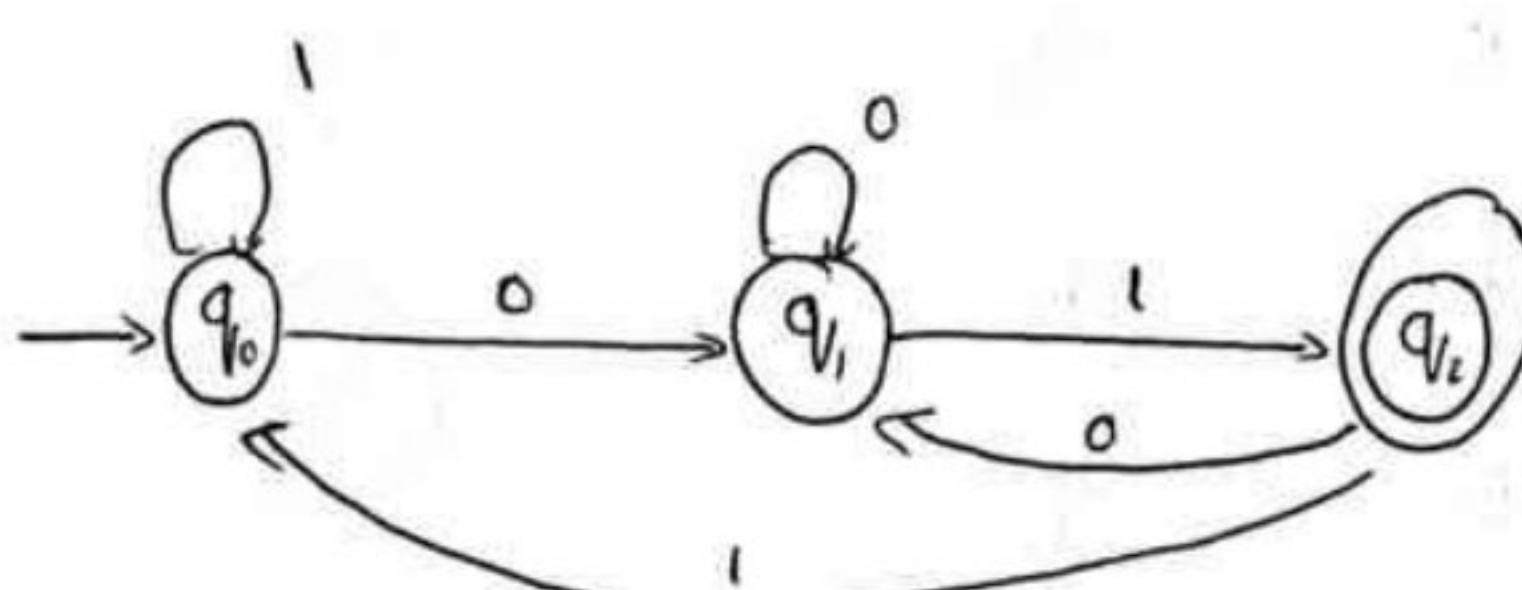
② Computability Theory :-

- * Whether the Problem can be solved or not

③ Complexity Theory :-

- * How much complexity can be added to machine

④ Draw the DFA for the language accepting strings ending with 01 over the alphabet $\Sigma = \{0, 1\}$



Note:- The outputs are not represented in the DFA Transition diagram.

$$M = \left\{ Q, \Sigma, \delta, q_0, F \right\}$$

where

- $Q = \{q_0, q_1, q_2\}$
- $\Sigma = \{0, 1\}$
- δ is defined as:
- $\delta(q_0, 0) = q_1$
- $\delta(q_0, 1) = q_0$
- $\delta(q_1, 0) = q_1$
- $\delta(q_1, 1) = q_2$
- $\delta(q_2, 0) = q_1$
- $\delta(q_2, 1) = q_0$

q_0 is the start state and q_2 is the accept state.

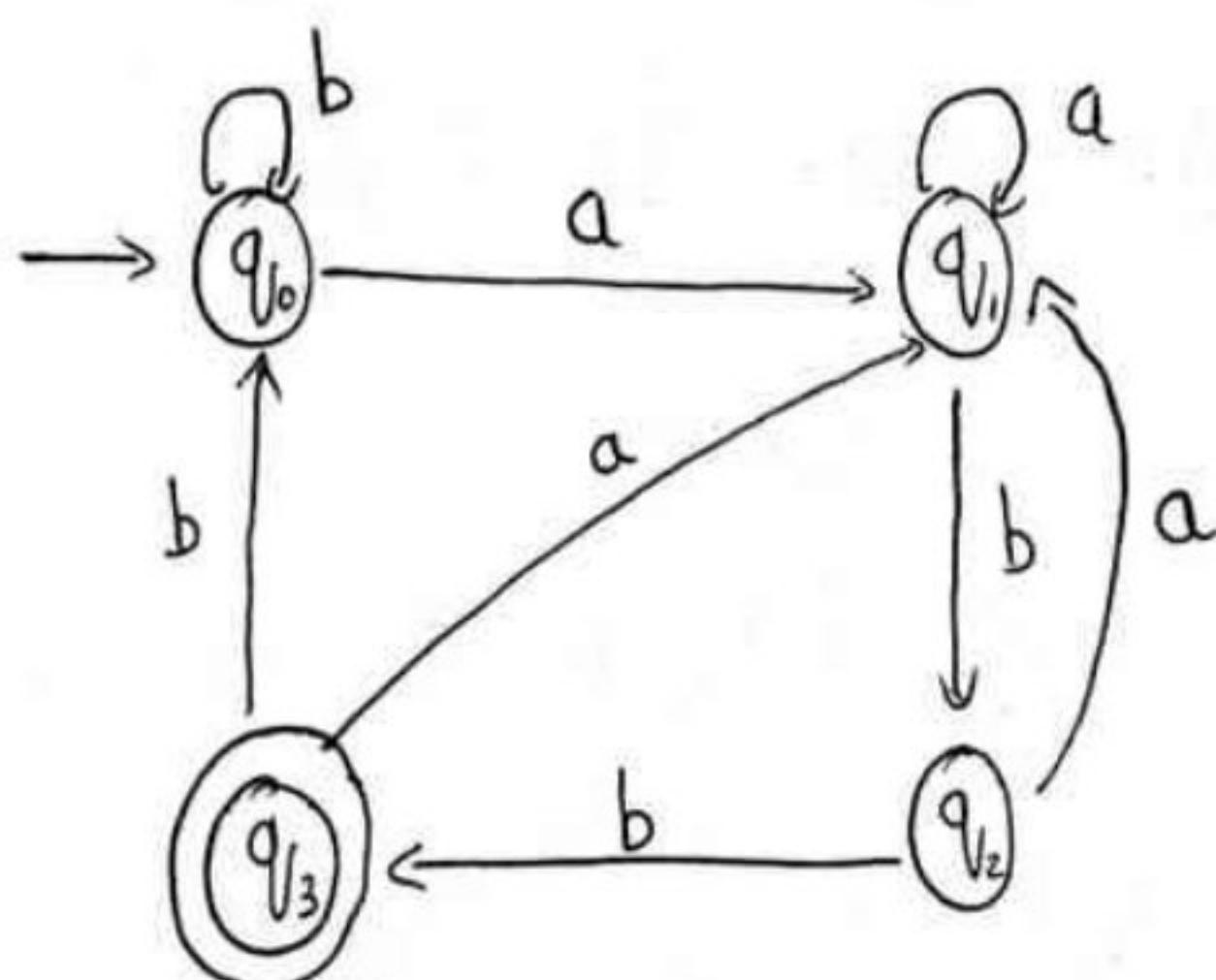
P. S	N. S	
	0	1
$\rightarrow q_0$	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_0

$$R = \{ (0+1)^* 01 \}$$

② Draw the DFA for the language accepting strings ending with 'abb' over the alphabet $\Sigma = \{0, b\}$.

The ending is abb, so we need 4 states

q_0, q_1, q_2, q_3



P. S	N. S	
	a	b
$\rightarrow q_0$	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_1	q_0

$$M = \left\{ Q, \Sigma, \delta, q_0, q_3 \right\}$$

\downarrow
 F

$$R = \{(a+b)^*abb\}$$

$$\delta(q_0, a) = q_1$$

$$\delta(q_2, a) = q_1$$

$$\delta(q_0, b) = q_0$$

$$\delta(q_2, b) = q_3$$

$$\delta(q_1, a) = q_1$$

$$\delta(q_3, a) = q_1$$

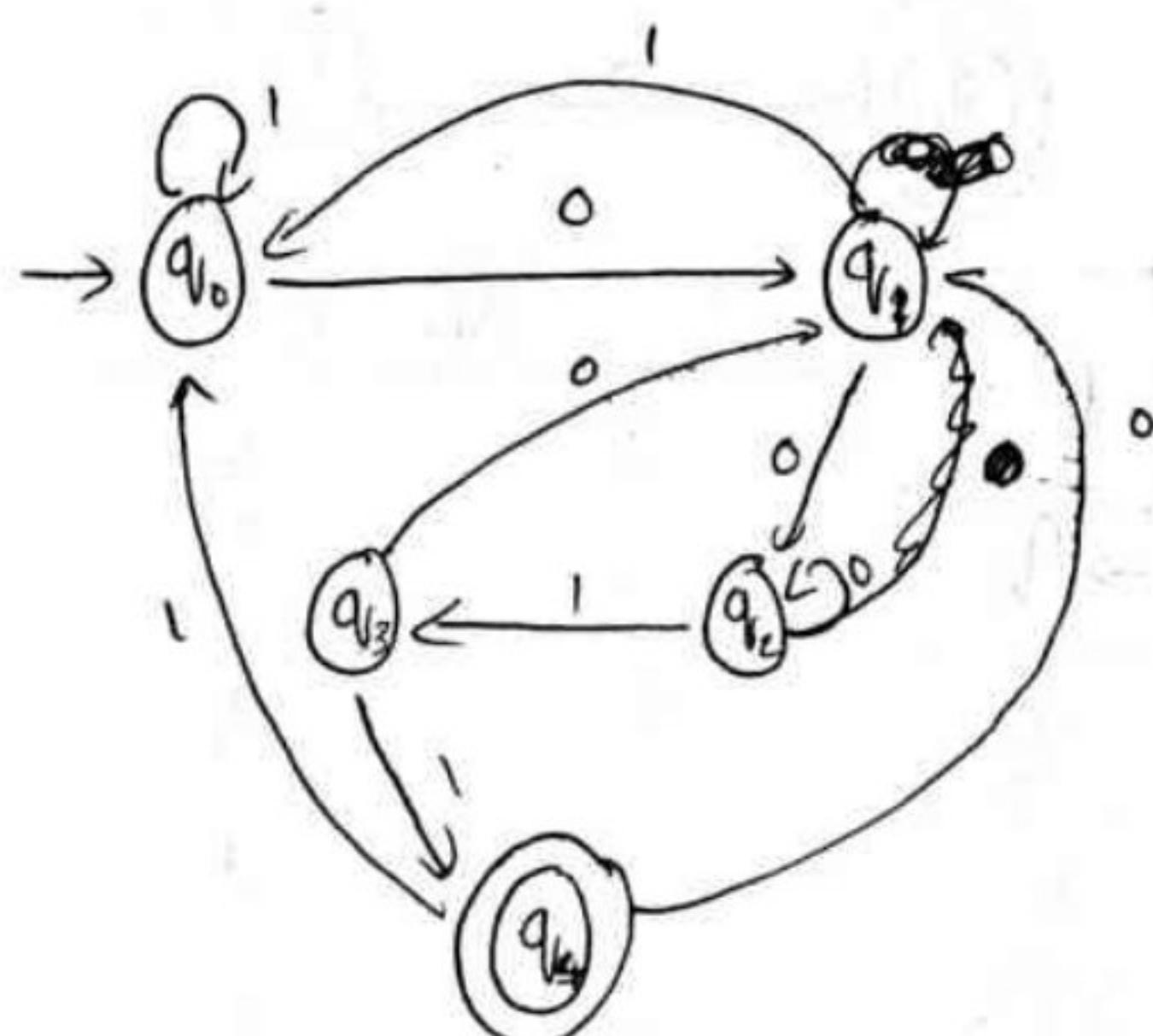
$$\delta(q_1, b) = q_2$$

$$\delta(q_3, b) = q_0$$

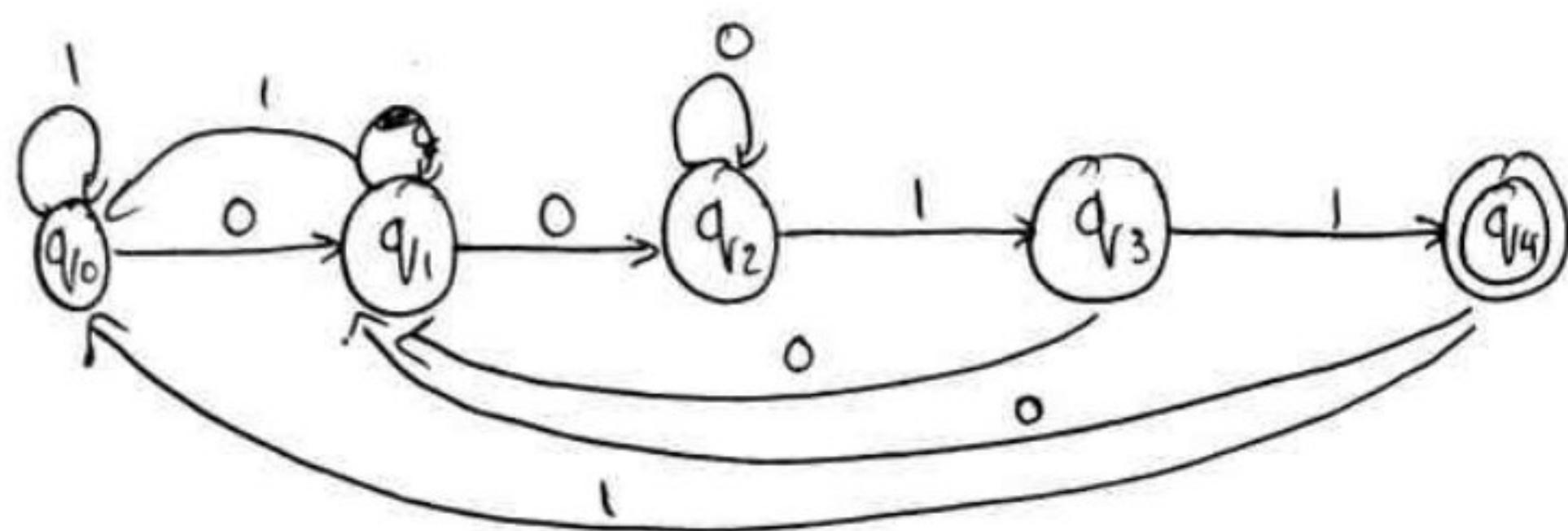
③ Draw the DFA for the language accepting strings ending with 0011 over the alphabet $\Sigma = \{0, 1\}$

We require 5 states.

$$\therefore Q = \{q_0, q_1, q_2, q_3, q_4\}$$



P. S	N. S	
	0	1
$\rightarrow q_0$	q_1	q_0
q_1	q_2	q_0
q_2	q_3	q_3
q_3	q_1	q_4
q_4	q_1	q_0



$$M = \left\{ Q, \Sigma, \delta, q_0, q_4 \right\}$$

$Q = \{q_0, q_1, q_2, q_3, q_4\}, \quad \Sigma = \{0, 1\}, \quad \delta = \text{transitions}, \quad q_0, q_4 \text{ are marked}\right\}$

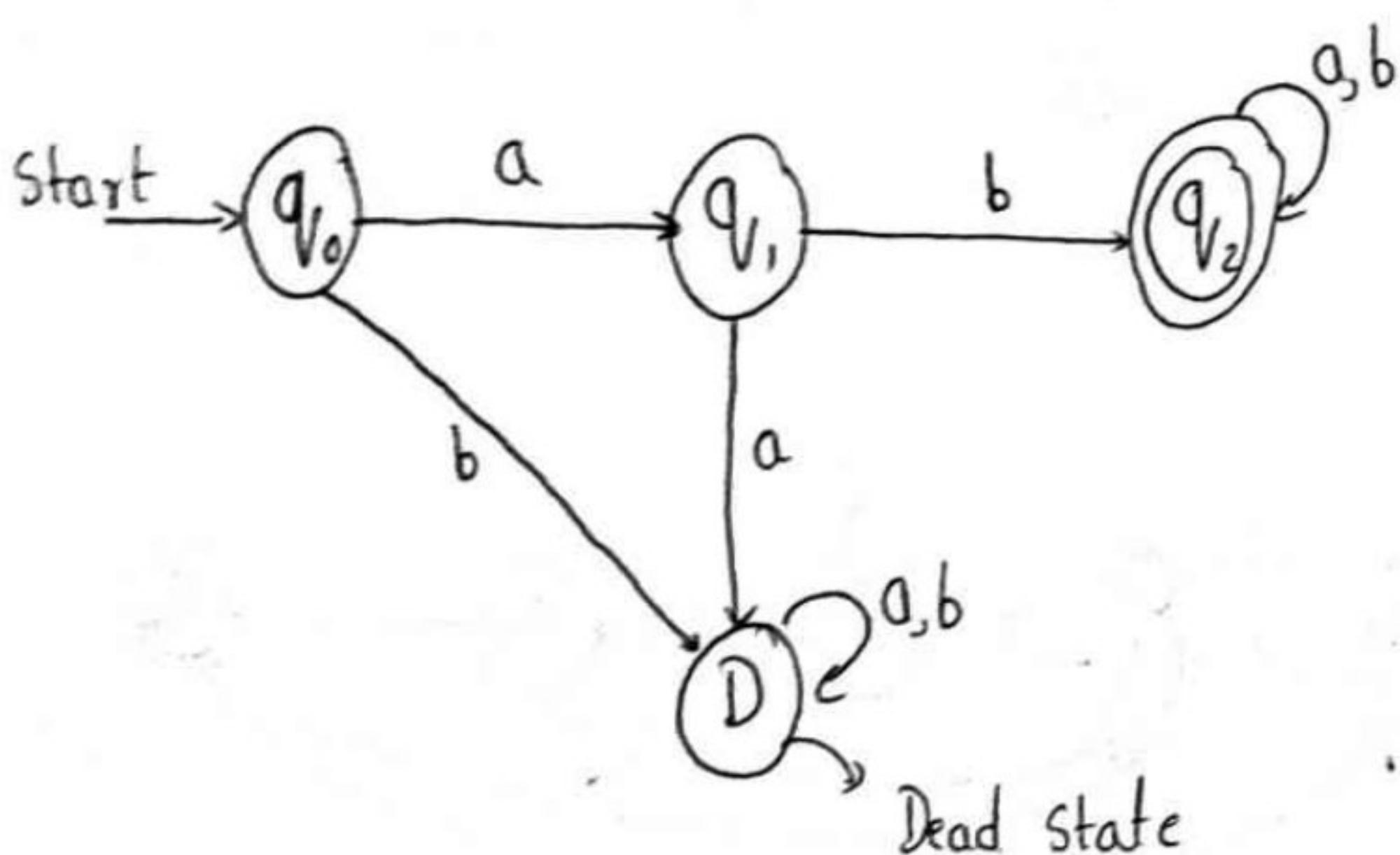
$$R = \{(0+1)^* 0011\}$$

④ Draw the DFA for the language accepting string starting with 'ab' over the alphabet $\Sigma = \{a, b\}$.

Regular Expression,

$$R = \{ab(a+b)^*\}$$

There are 2 bits so we need four states



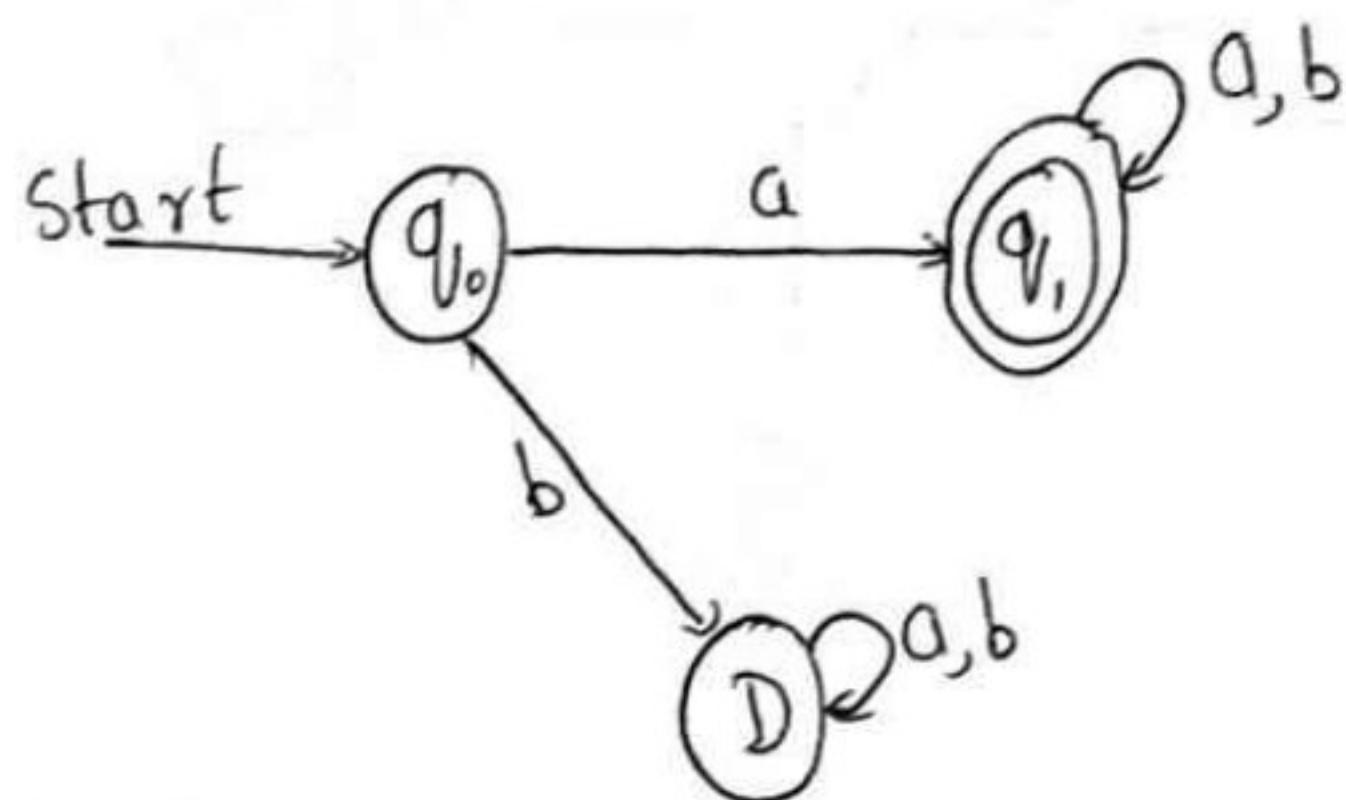
		N. S	
P. S		a	b
$\rightarrow q_{V_0}$		q_{V_1}	D
q_{V_1}		D	q_{V_2}
$(q_{V_2})^*$		q_{V_2}	q_{V_2}

⑤ Draw the DFA for the language accepting strings starting with 'a' for the input alphabet, $\Sigma = \{a, b\}$

Regular Expression,

$$R = \{ a (a+b)^* \}$$

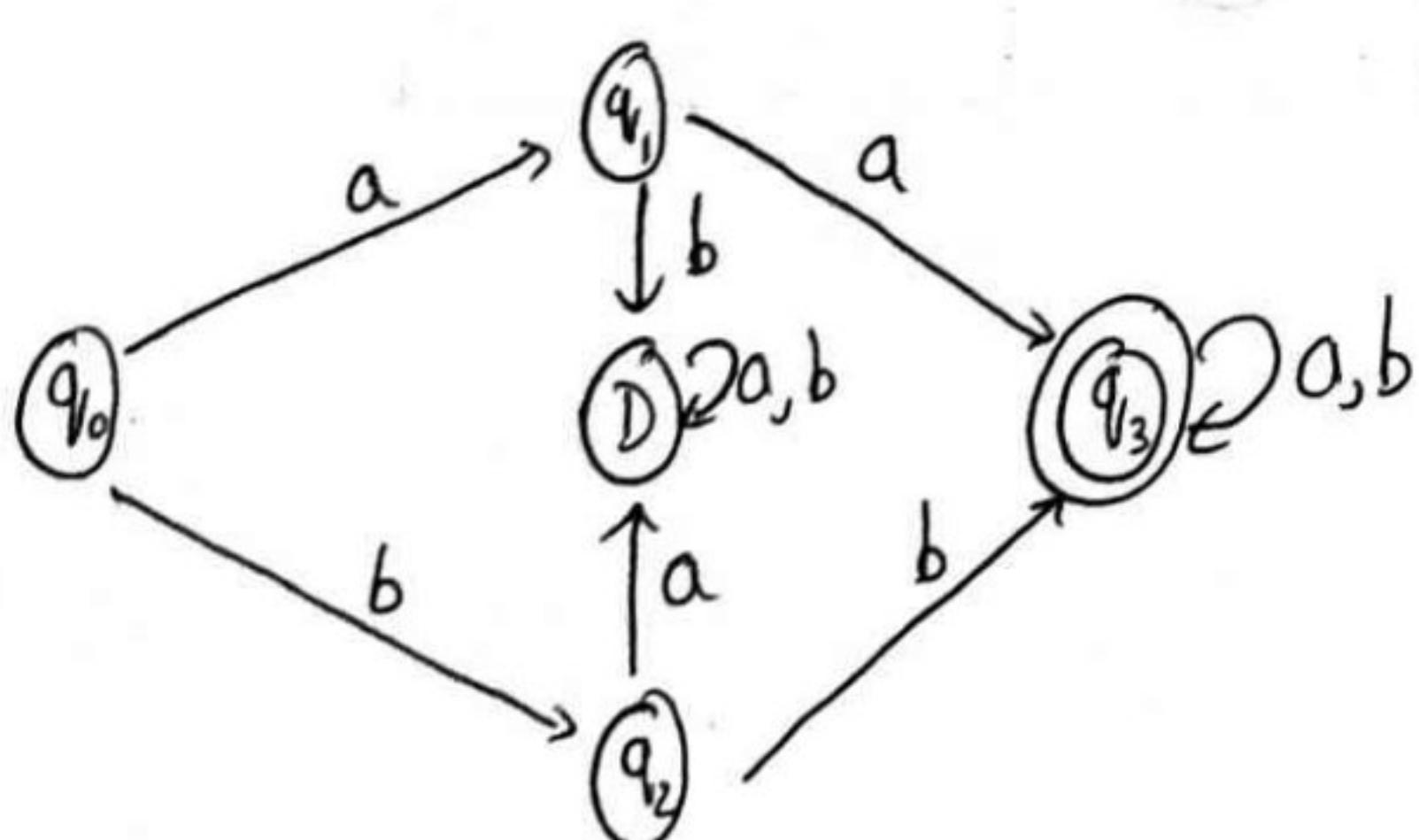
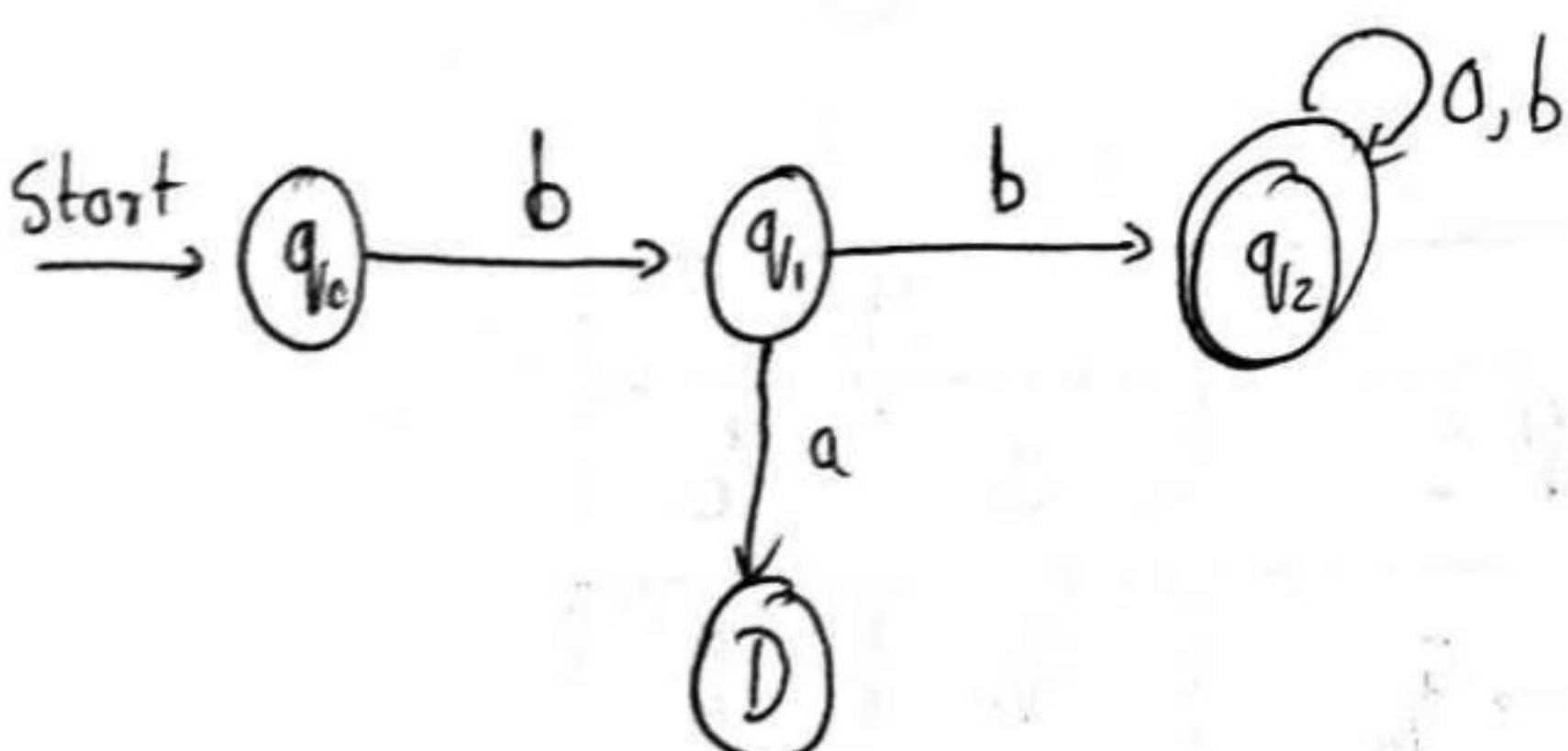
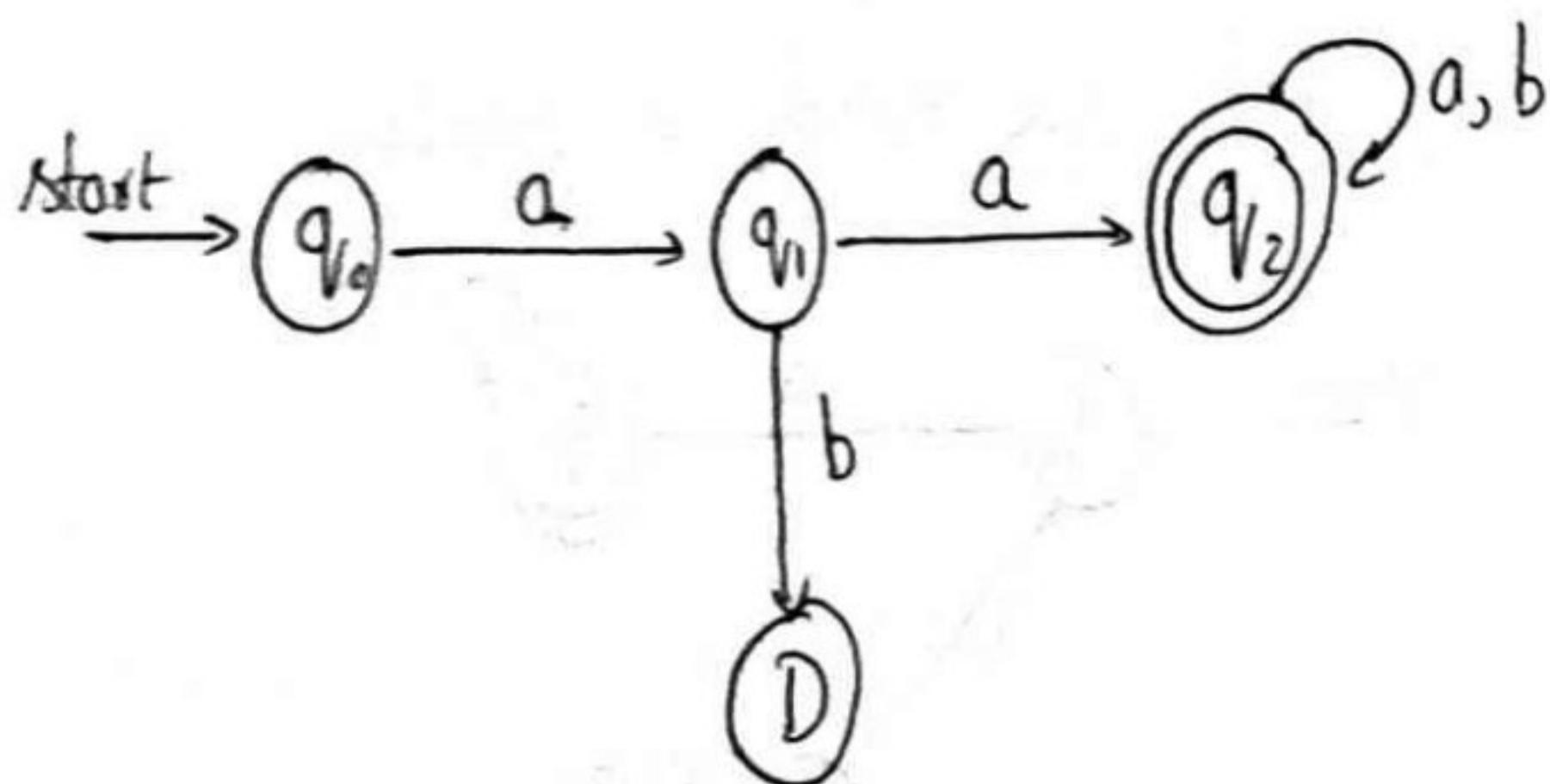
For '1' bit we need 3 states



	N.S.	
P.S	a	b
$\rightarrow q_{v_0}$	q_{v_1}	D
$(q_{v_1})^*$	q_{v_1}	q_{v_1}

⑥ Construct a DFA that accepts a language over the alphabet $\Sigma = \{a, b\}$ such that 'L' is a set of all strings starting with aa or bb

Total no. of states = 4



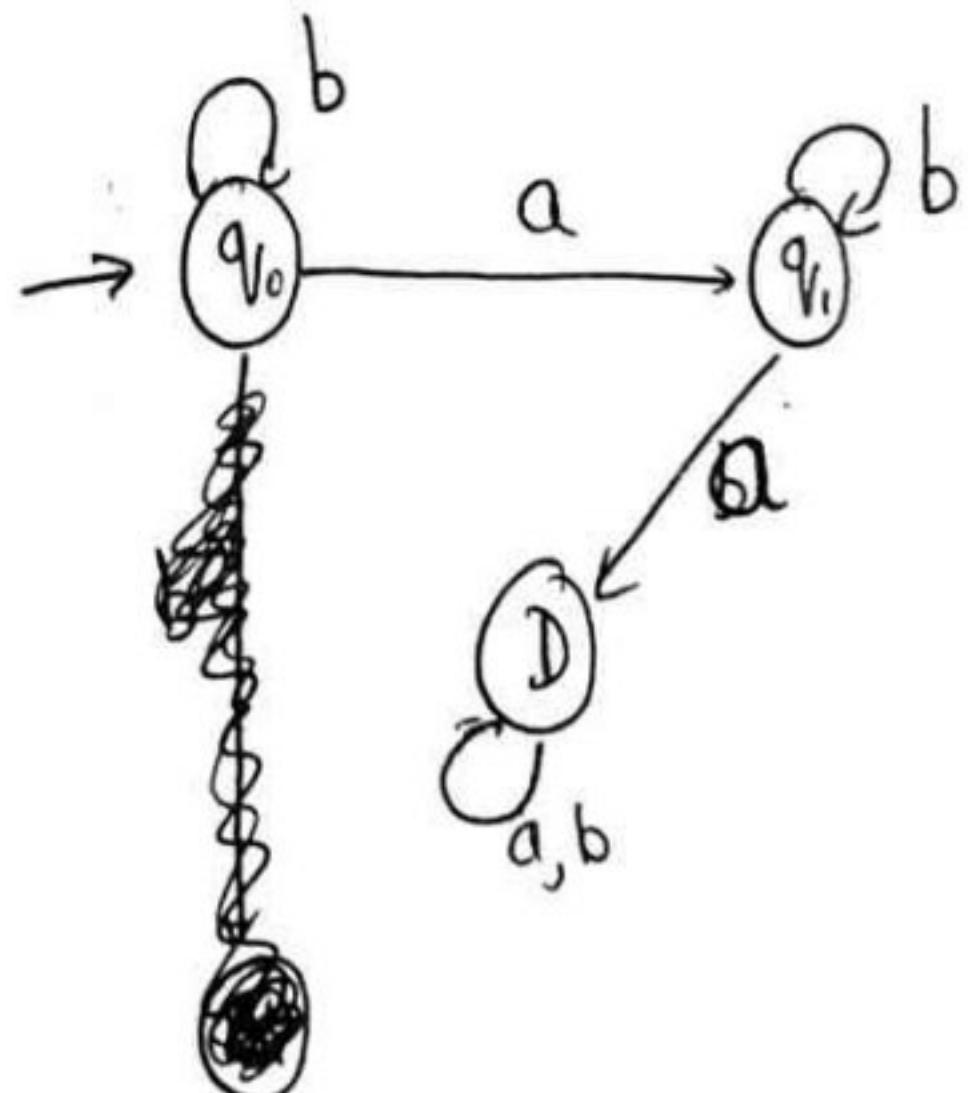
$$R = \{ (aa+bb)^* \}$$

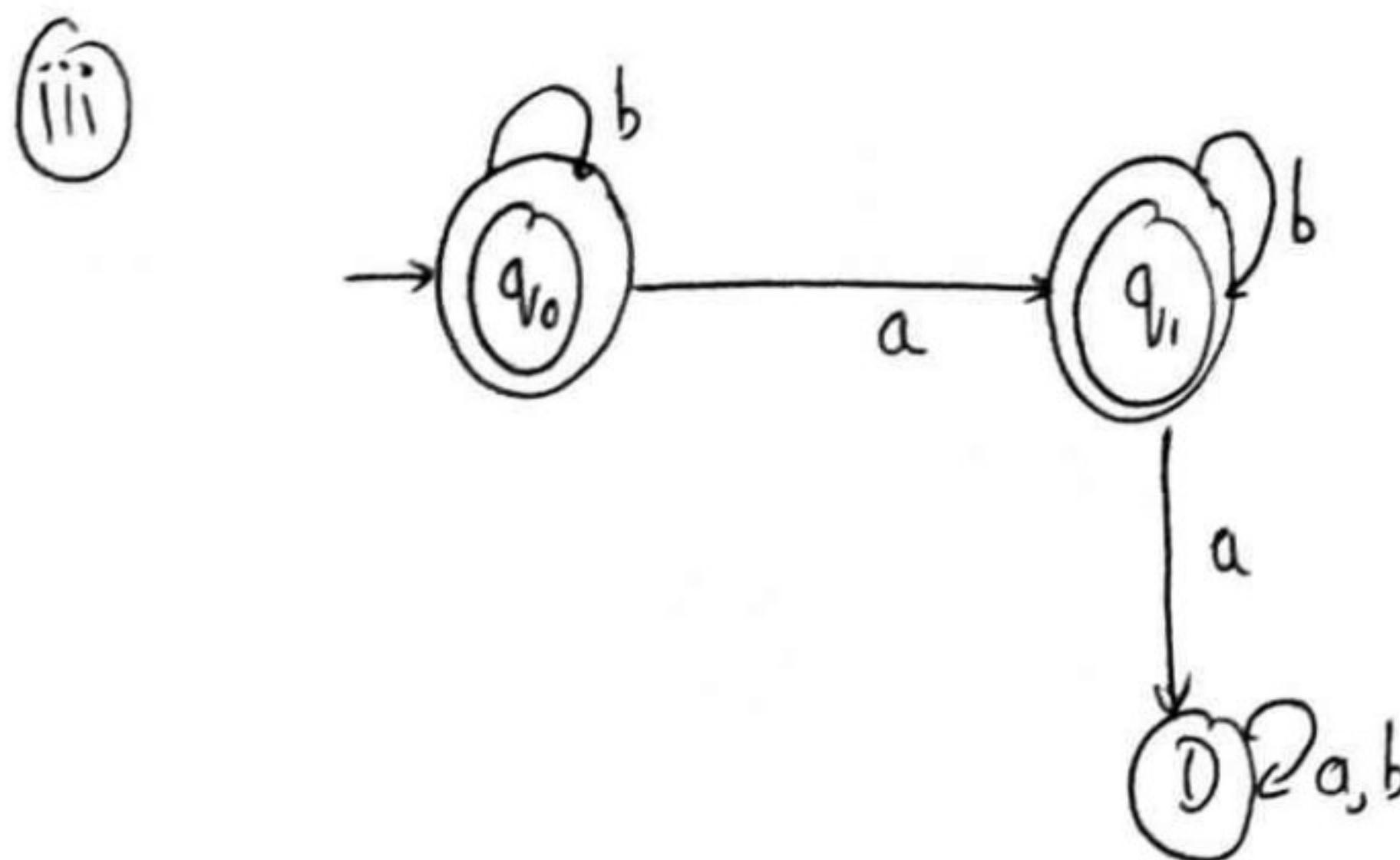
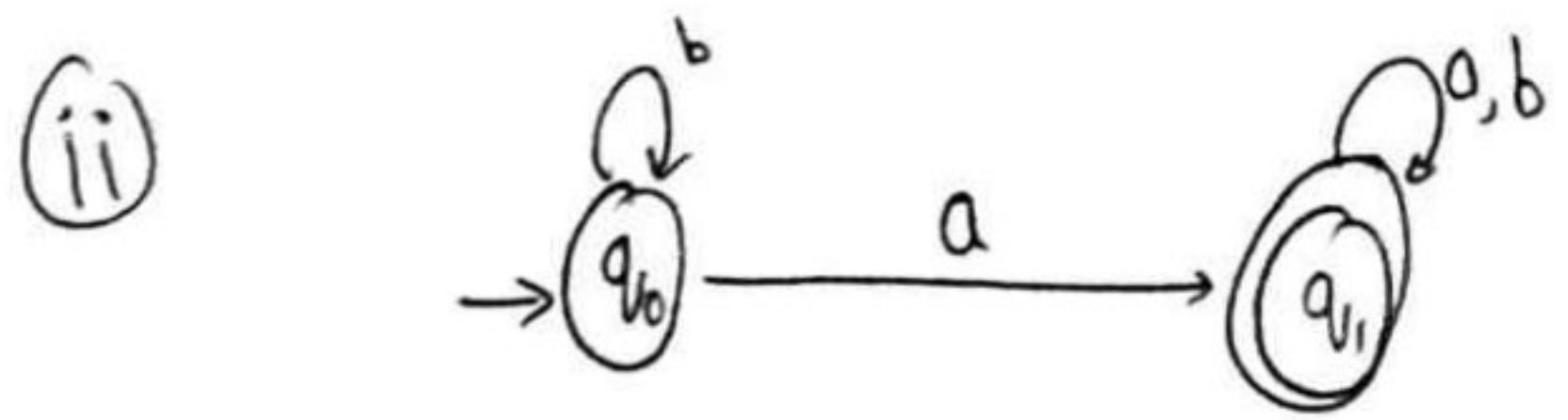
		N. S	
P. S	a	b	
q_{v_0}	q_{v_1}	q_{v_2}	
q_{v_1}	q_{v_3}	D	
q_{v_2}	D	q_{v_3}	
q_{v_3}	q_{v_3}	q_{v_3}	

⑦ Design the DFA that accepts the language 'L' over the alphabet $\Sigma = \{a, b\}$

- i exactly one 'a'
- ii atleast one 'a'
- iii atmost one 'a'

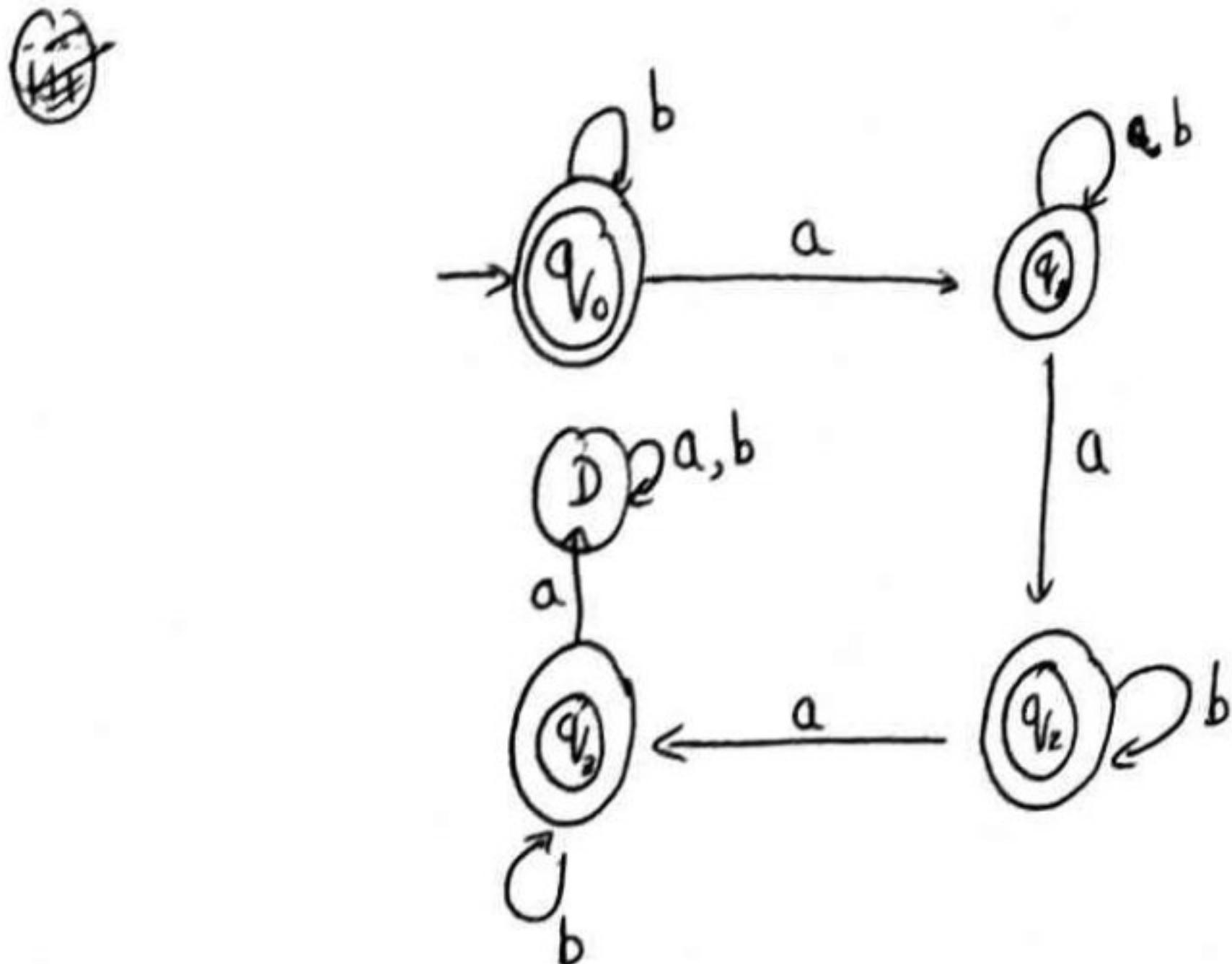
i



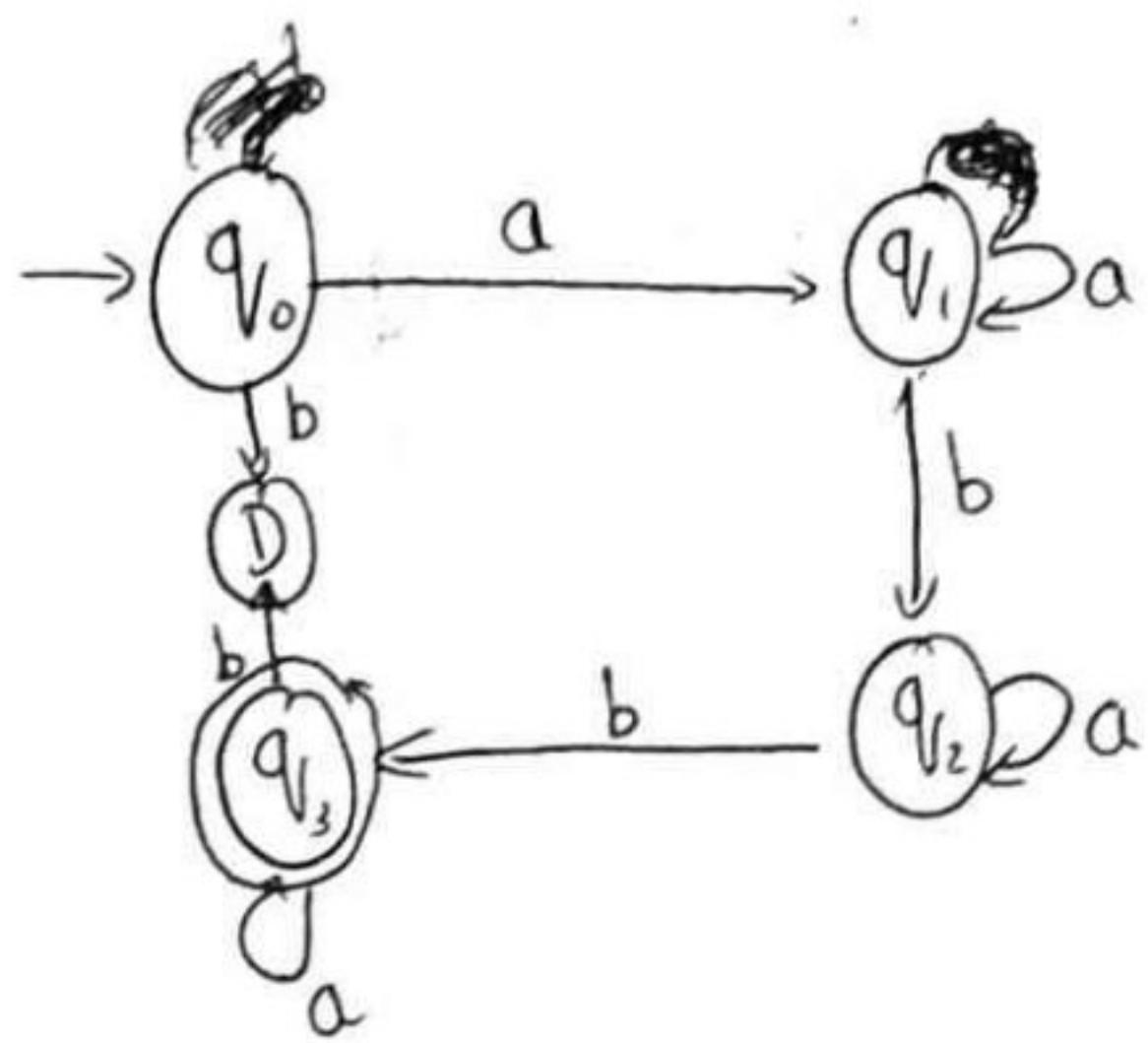


⑦ Design a DFA that accepts a language 'L' over the alphabet $\Sigma = \{a, b\}$ such that 'L' is a set of string

- i) no more than 3 a
- ii) at least a & exactly two b's



(ii)



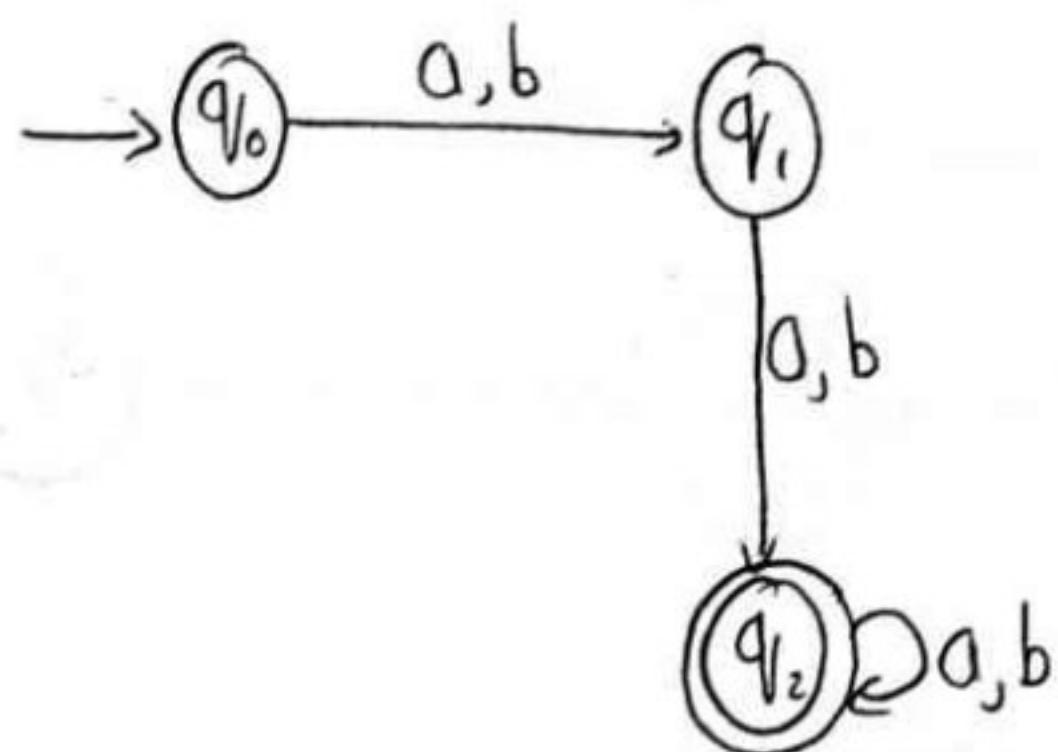
⑥ Construct a DFA that accepts a language 'L' over the language alphabet $\Sigma = \{a, b\}$ such that

i) length of string is atleast 2

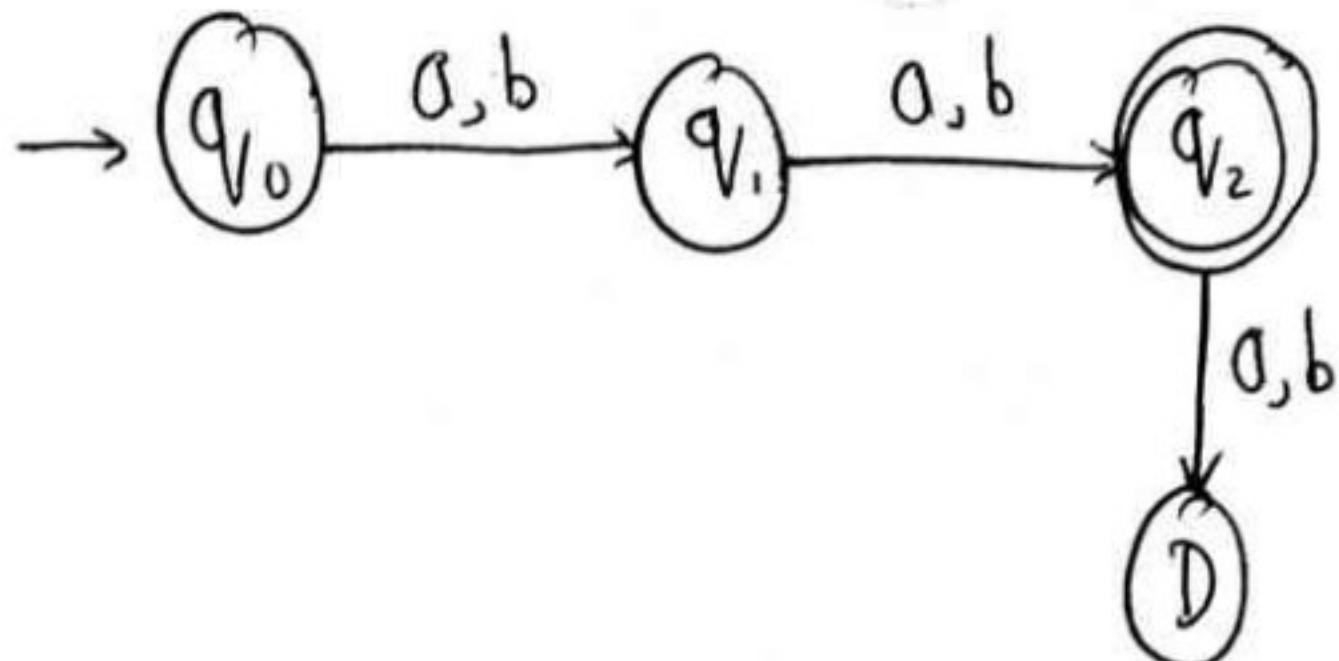
ii) Length of string is exactly 2

iii) Length of string is atmost 2

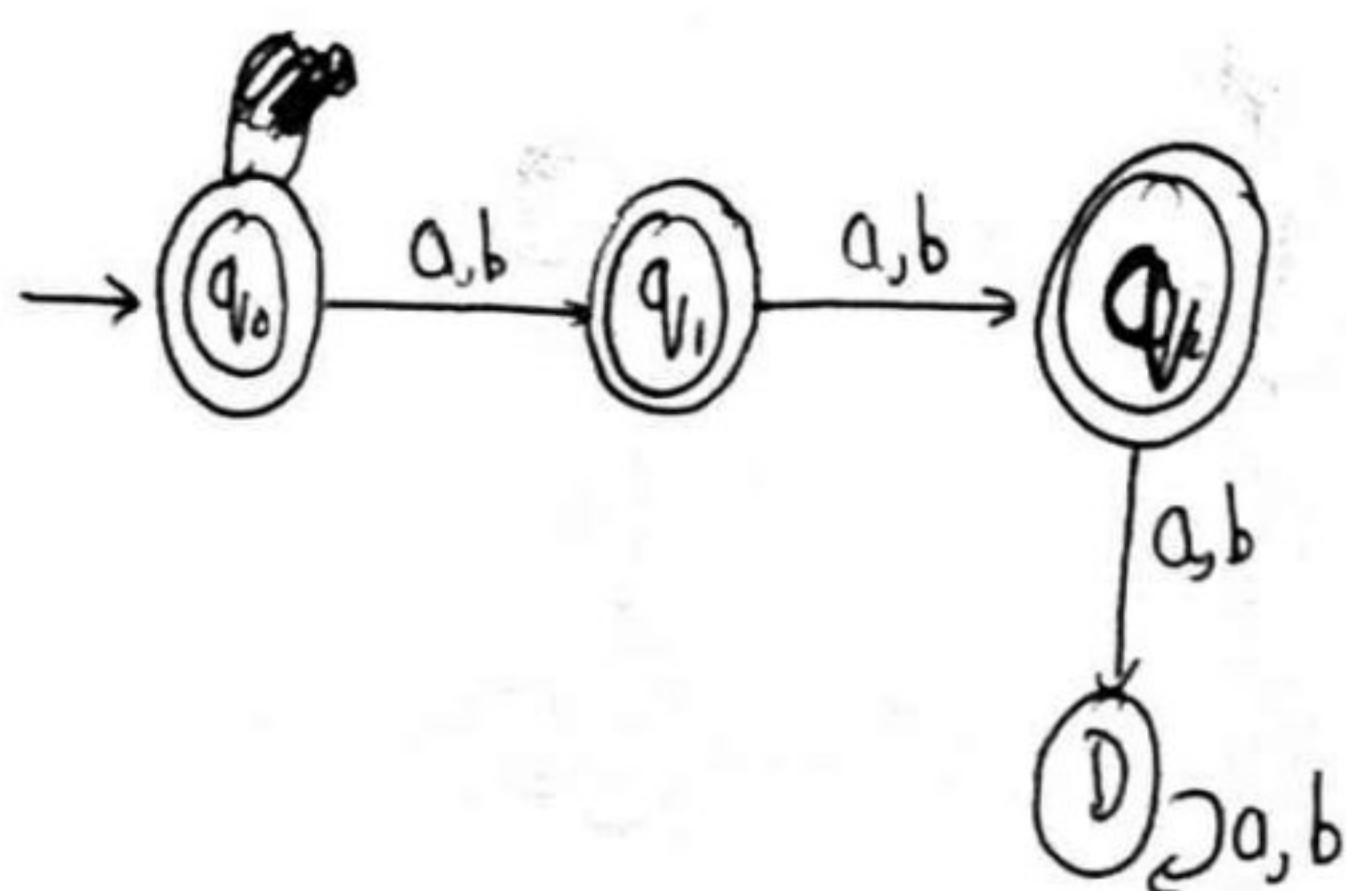
(i)



(ii)



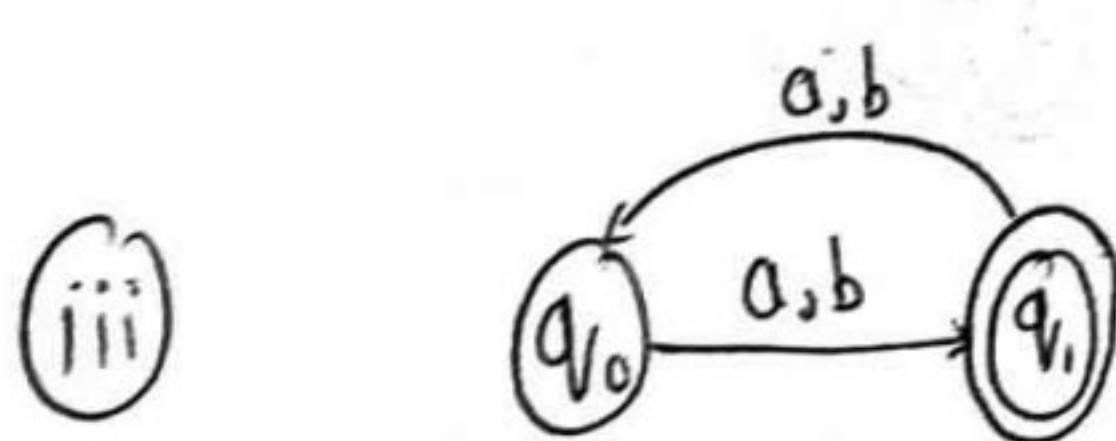
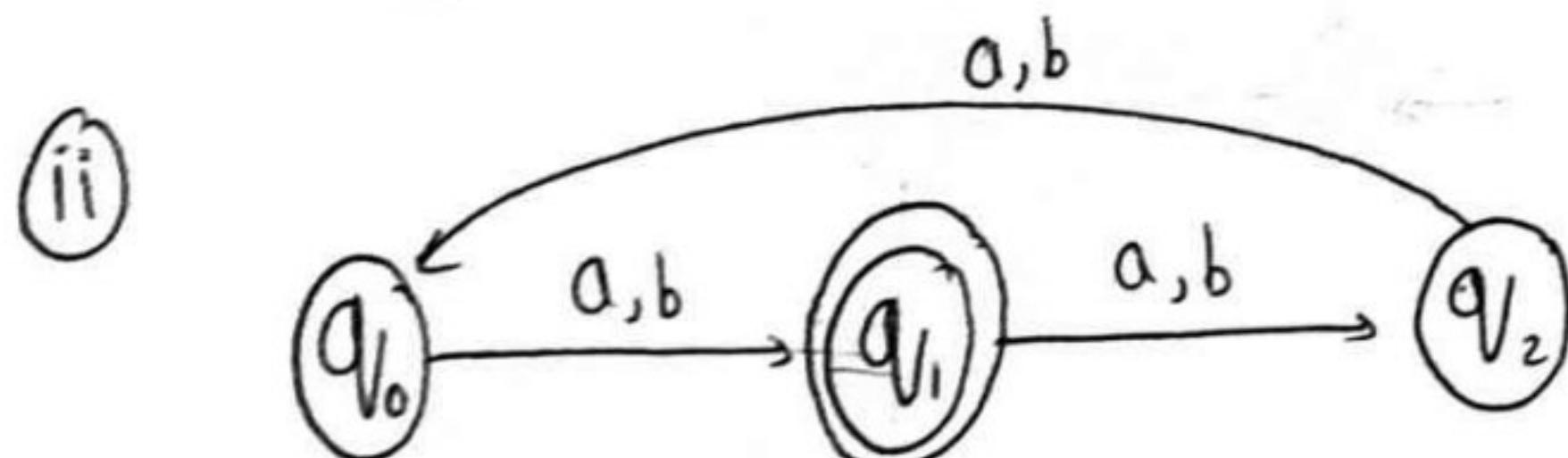
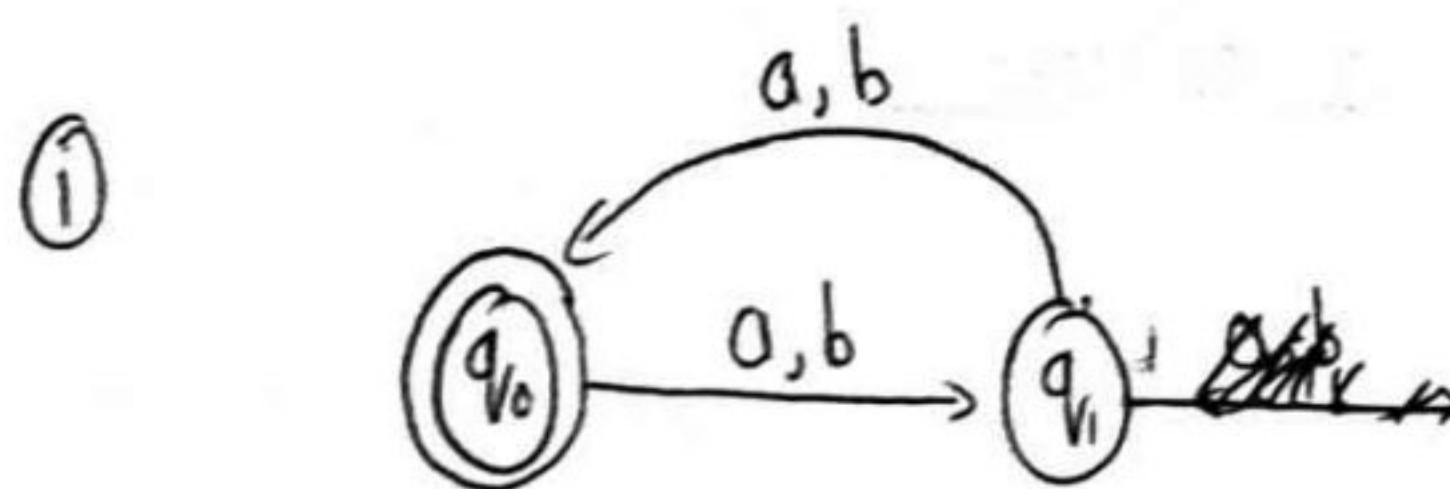
(iii)



④ Construct a DFA that accepts a string over input alphabet $\Sigma = \{a, b\}$ such that

i) $|w| \bmod 2 = 0$ ($aa, abab, \dots$)

ii) $(w \bmod 3 = 1)$ iii) $(w \bmod 2 = 1)$



\Rightarrow Non-Deterministic Finite Automata (NFA);-

A NFA is defined by quintuple

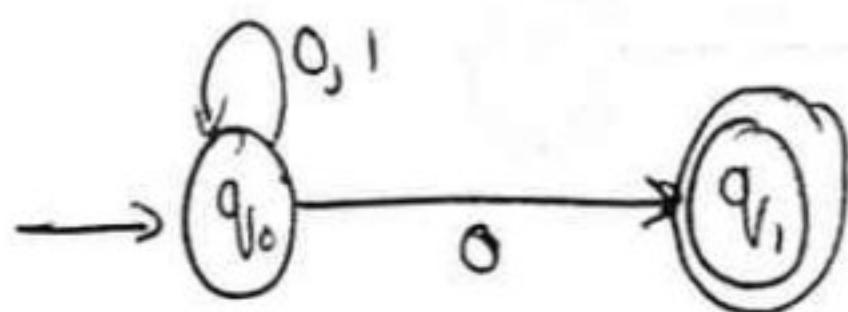
$$M = (\emptyset, \Sigma, \delta, q_0, F)$$

where,

$$\delta: \emptyset \times \Sigma \rightarrow 2^{\emptyset}$$

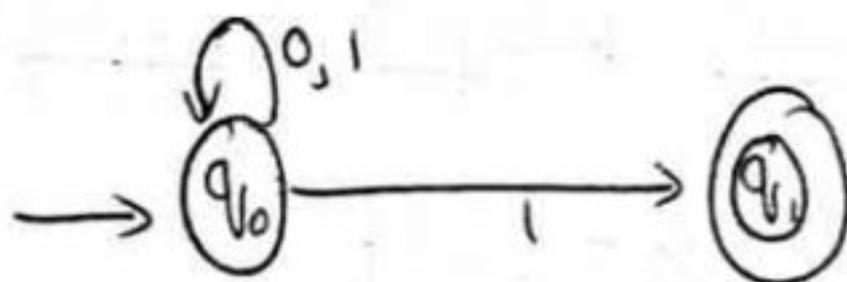
- ① Construct an NFA that accepts all set of strings ending with '0' over an input alphabet $\Sigma = \{0, 1\}$.

$$R = \{(0+1)^* 0\}$$

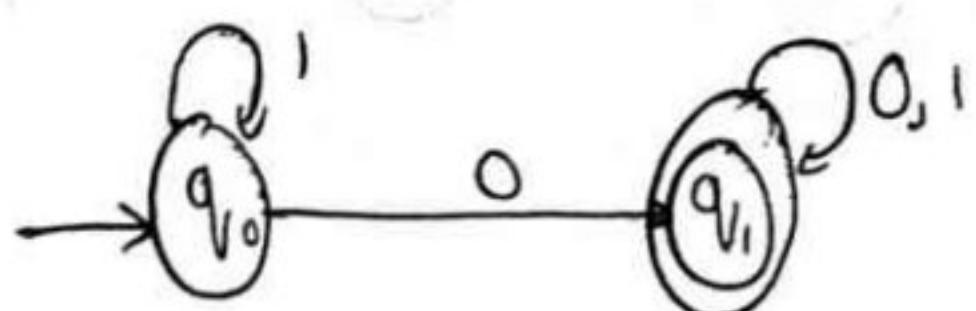


- ② Construct an NFA for following cases:-

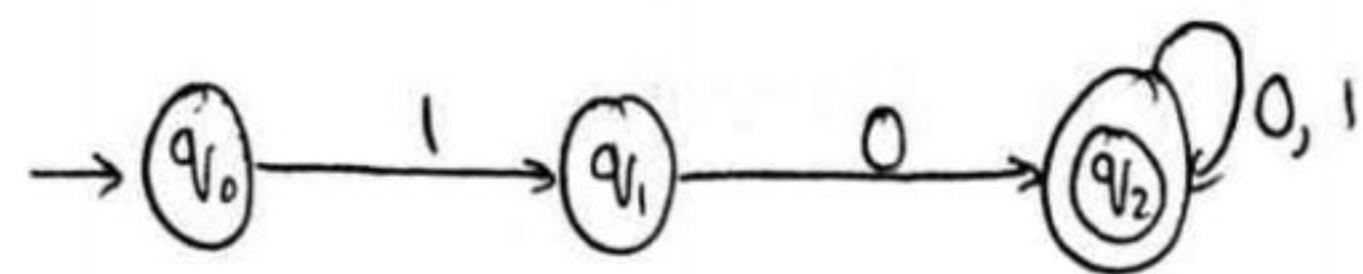
- i) L_1 = Set of all strings ending with '1'



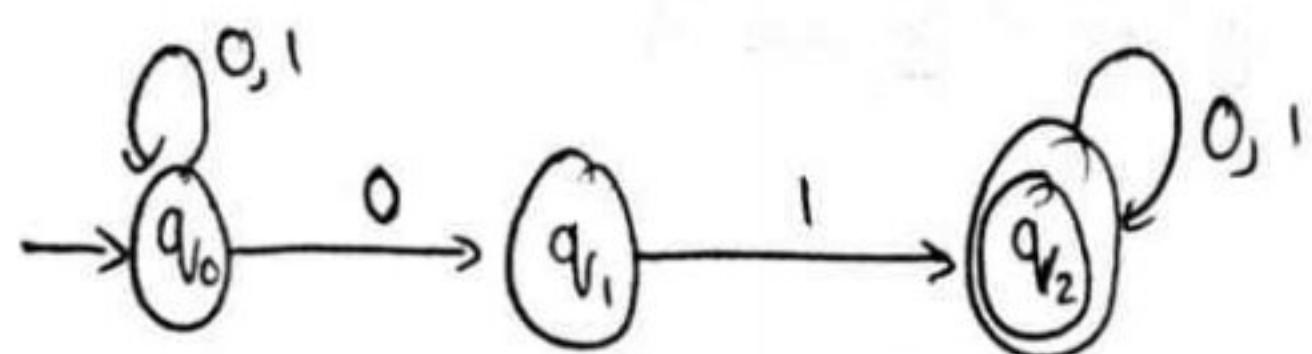
- ii) L_2 = Set of all strings that contain '0'



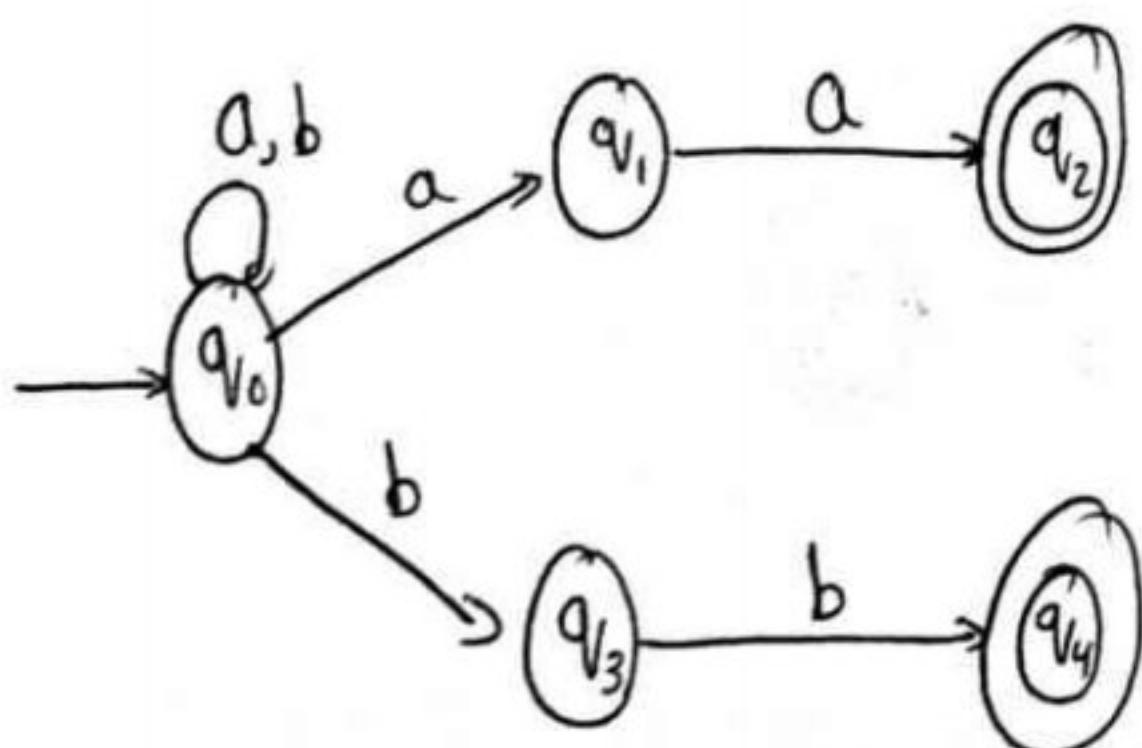
iii) L_3 = Set of all String that starts with 10



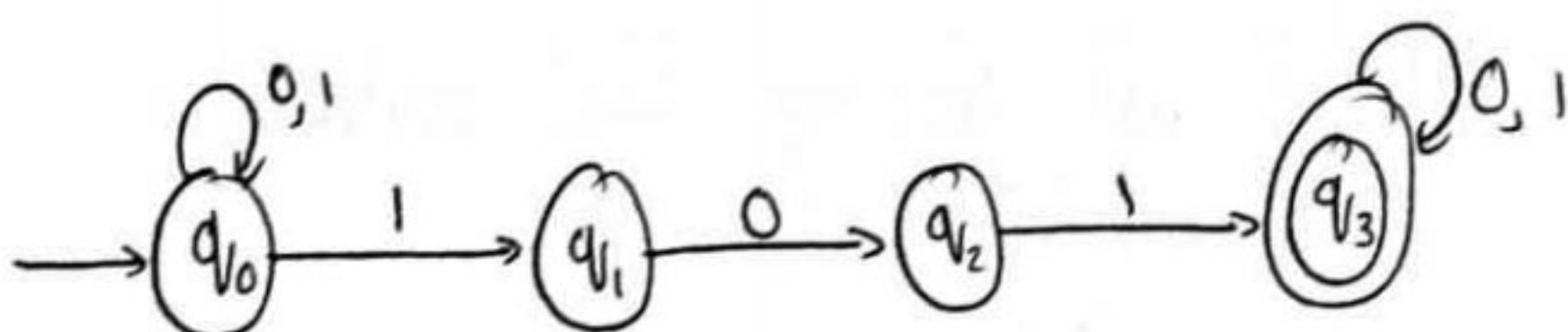
iv) L_4 = Set of all Strings that contain 01



③ Construct an NFA that accepts strings ending with aa@bb



④ Construct an NFA that accepts strings containing 101 as Substring over alphabet $\Sigma = \{0,1\}$

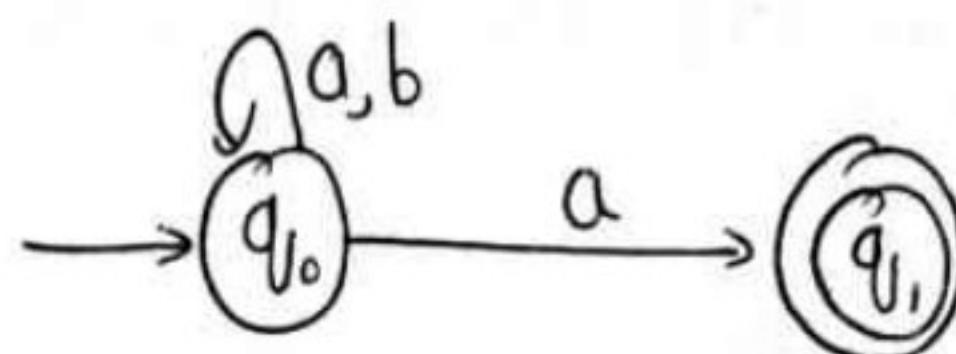


→ Conversion of NFA to DFA :-

① Convert from NFA to DFA, the language is set of all strings over $\Sigma = \{a, b\}$ that ends with 'a'.

→ Subset Construction Method :-

The NFA is



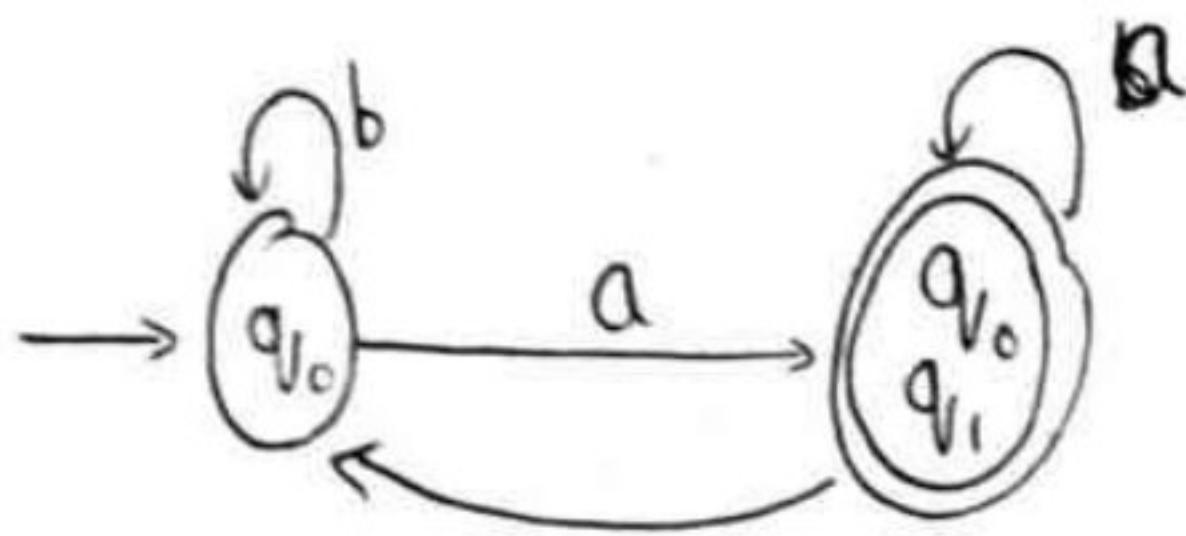
Transition Table :-

	a	b
$\rightarrow q_0$	$[q_0, q_1]$	q_0
q_1	\emptyset	\emptyset

↓ Considering q_0, q_1 as single state
 q_0q_1

	a	b
$\rightarrow q_0$	$[q_0q_1]$	q_0
q_0q_1	$[q_0q_1]$	q_0

\therefore The DFA is



② Construct a DFA equivalent Machine A

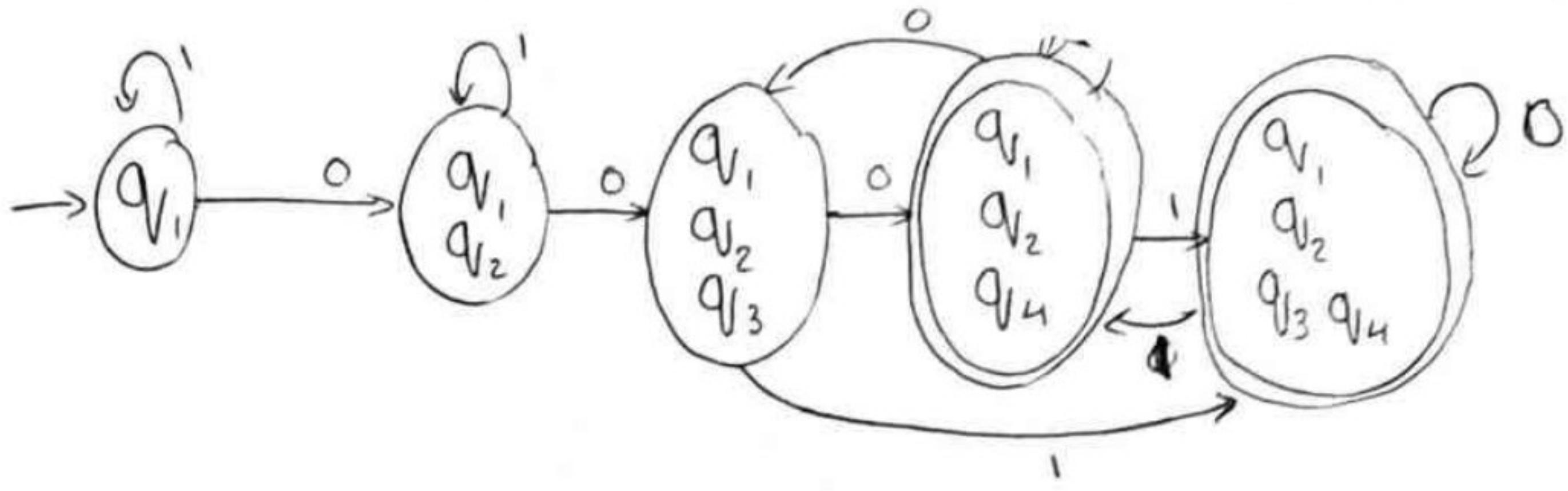
$$M = \left(\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \delta, q_1, q_4 \right)$$

where, δ is given as

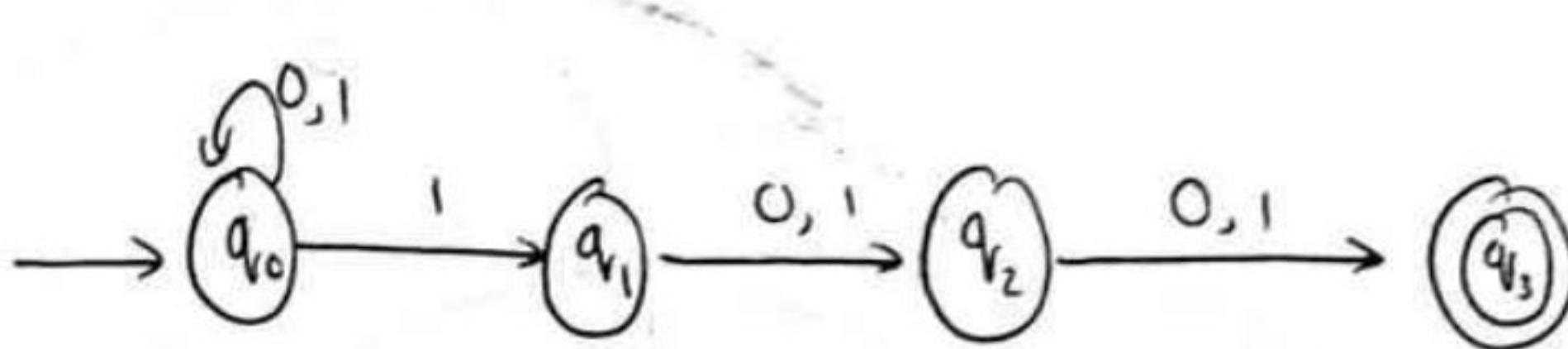
	0	1
$\rightarrow q_1$	q_1, q_2	q_1
q_2	q_3	q_2
q_3	q_4	q_4
$* q_4$	-	q_4

↓

	0	1
$\rightarrow q_1$	$[q_1, q_2]$	q_1
q_1, q_2	$[q_1, q_2, q_3]$	$[q_1, q_2]$
q_1, q_2, q_3	$[q_1, q_2, q_3, q_4]$	$[q_1, q_2, q_4]$
$* q_1, q_2, q_3, q_4$	$[q_1, q_2, q_3, q_4]$	$[q_1, q_2, q_4]$
$* q_1, q_2, q_4$	$[q_1, q_2, q_3, q_4]$	$[q_1, q_2, q_4]$



- ③ Convert NFA to equivalent DFA for the transition Diagram shown below :-

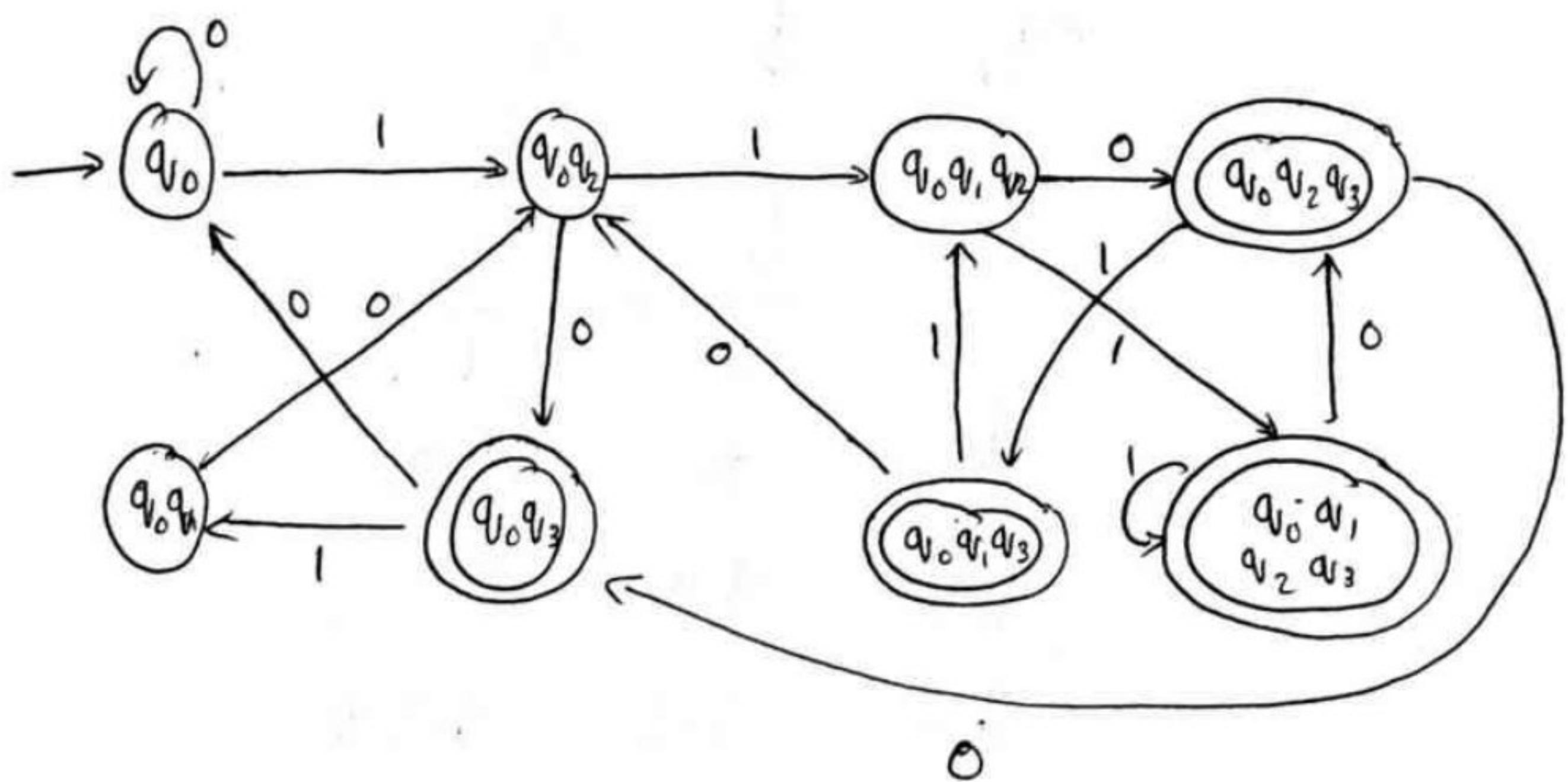


	0	1
$\rightarrow q_{V_0}$	$q_{V_0} q_{V_1}$	$[q_{V_0}, q_{V_1}]$
q_{V_1}	q_{V_2}	q_{V_2}
q_{V_2}	q_{V_3}	q_{V_3}
$* q_{V_3}$	\emptyset	\emptyset

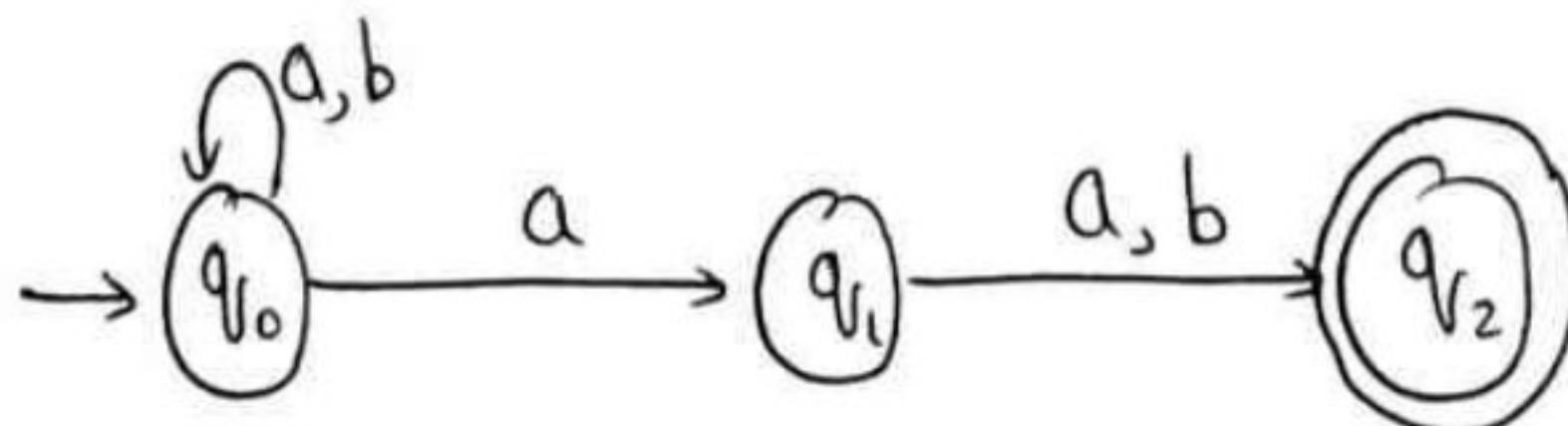
↓

	0	1
q_{V_0}	q_{V_0}	$q_{V_0} q_{V_1}$
$q_{V_0} q_{V_1}$	$q_{V_0} q_{V_2}$	$q_{V_0} q_{V_1} q_{V_2}$
$q_{V_0} q_{V_2}$	$q_{V_0} q_{V_3}$	$q_{V_0} q_{V_1} q_{V_3}$

	0	1
* $q_{V_0} q_{V_3}$	q_{V_0}	$q_{V_0} q_{V_1}$
$q_{V_0} q_{V_1} q_{V_2}$	$q_{V_0} q_{V_2} q_{V_3}$	$q_{V_0} q_{V_1} q_{V_2} q_{V_3}$
* $q_{V_0} q_{V_1} q_{V_3}$	$q_{V_0} q_{V_3}$	$q_{V_0} q_{V_1} q_{V_2}$
* $q_{V_0} q_{V_2} q_{V_3}$	$q_{V_0} q_{V_3}$	$q_{V_0} q_{V_1} q_{V_3}$
* $q_{V_0} q_{V_1} q_{V_2} q_{V_3}$	$q_{V_0} q_{V_2} q_{V_3}$	$q_{V_0} q_{V_1} q_{V_2} q_{V_3}$

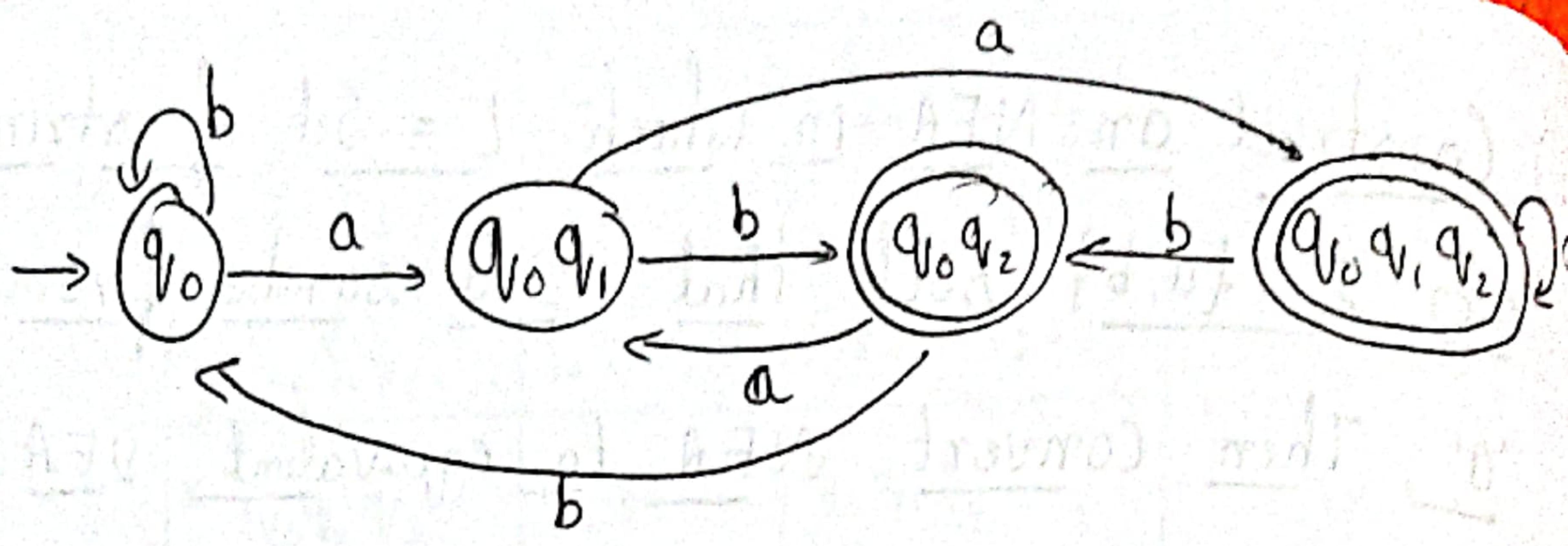


④ Construct an NFA in which $L = \text{Set of strings}$ over $\Sigma = \{a, b\}$ such that 2nd symbol from RHS is '0'. Then convert NFA to equivalent DFA.



Transition Table :-

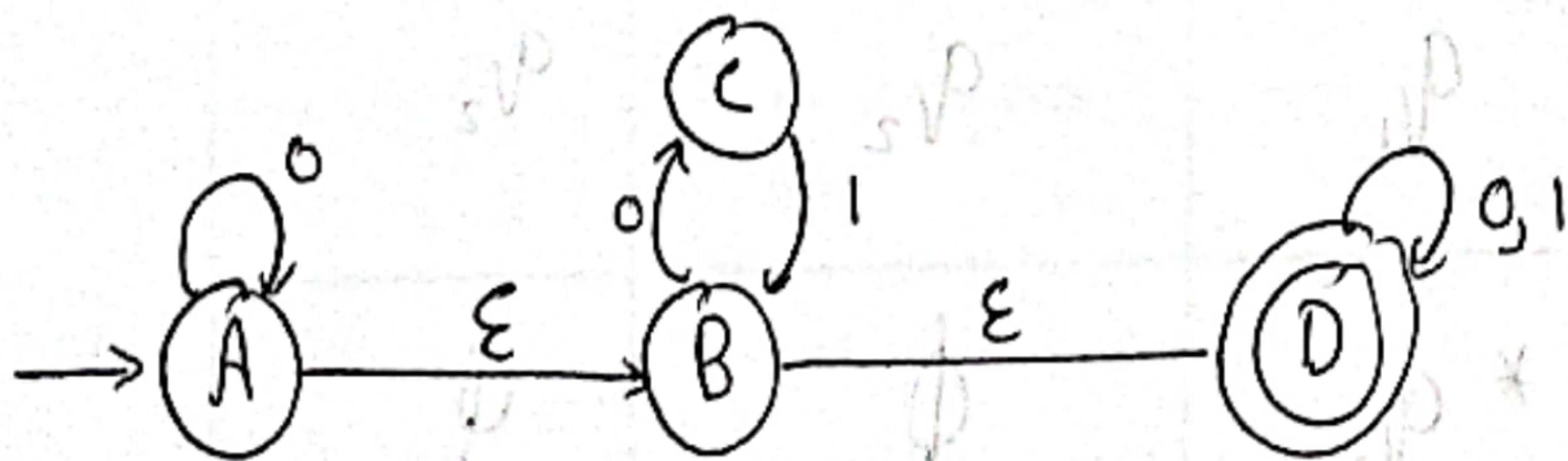
	a	b
$\rightarrow q_{v_0}$	$[q_{v_0} q_{v_1}]$	q_{v_0}
q_{v_1}	q_{v_2}	q_{v_2}
$* q_{v_2}$	\emptyset	\emptyset
	a	b
$\rightarrow q_{v_0}$	$[q_{v_0} q_{v_1}]$	q_{v_0}
$q_{v_0} q_{v_1}$	$[q_{v_0} q_{v_1} q_{v_2}]$	$q_{v_0} q_{v_2}$
$* q_{v_0} q_{v_2}$	$[q_{v_0} q_{v_1}]$	q_{v_0}
$* q_{v_0} q_{v_1} q_{v_2}$	$[q_{v_0} q_{v_1} q_{v_2}]$	$[q_{v_0} q_{v_2}]$



$\Rightarrow \epsilon\text{-NFA to NFA}$ (conversion:-)

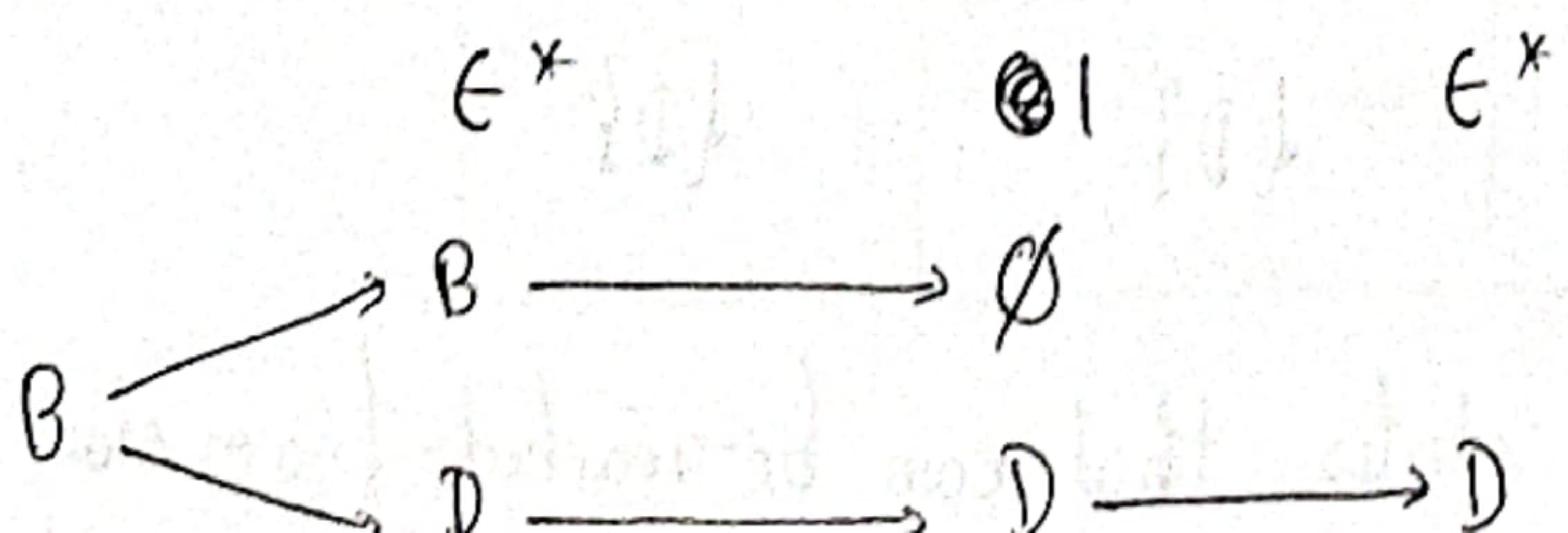
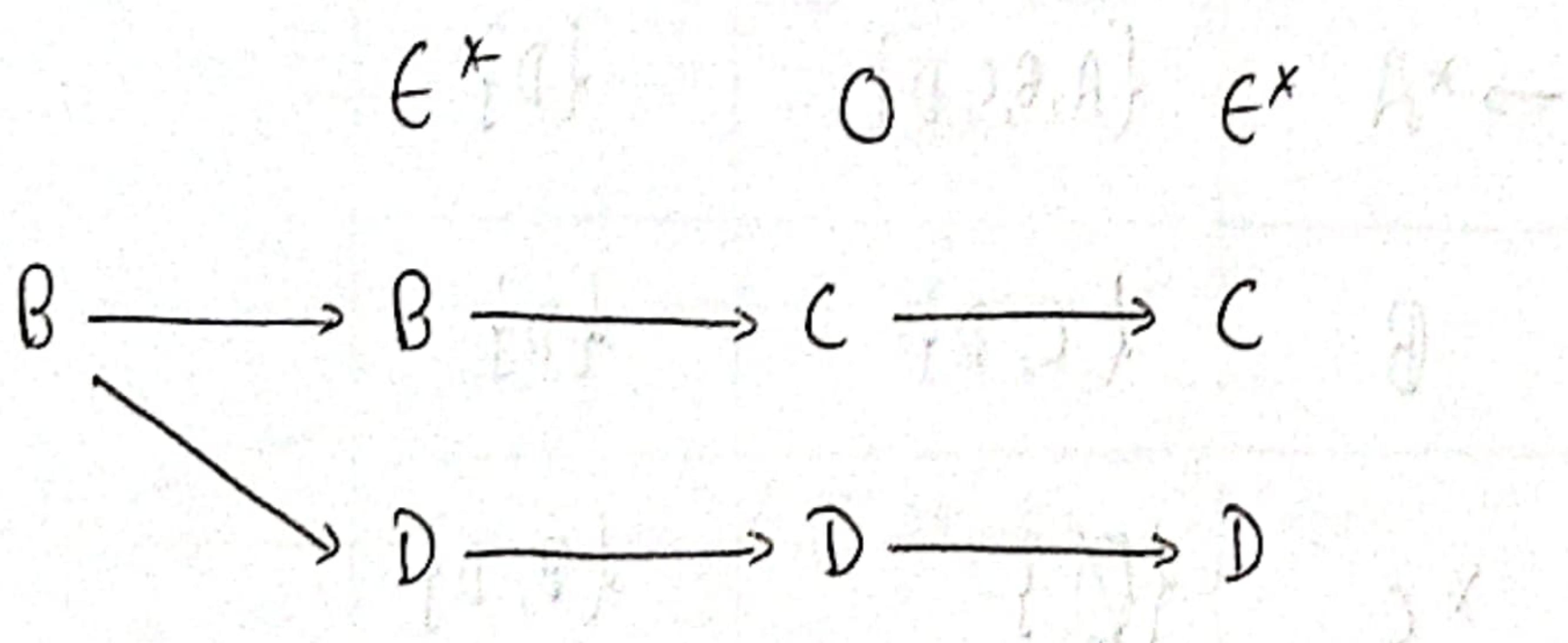
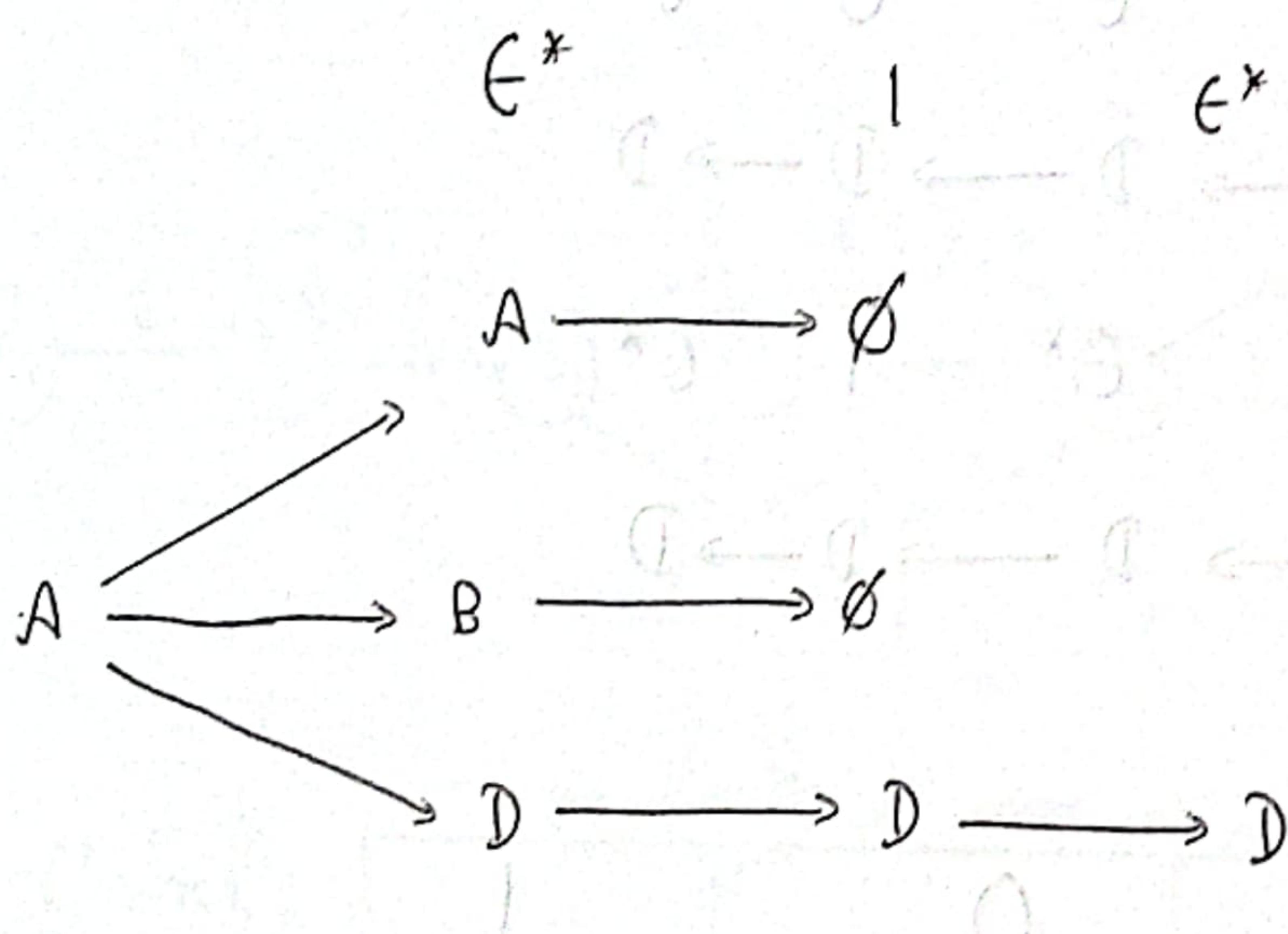
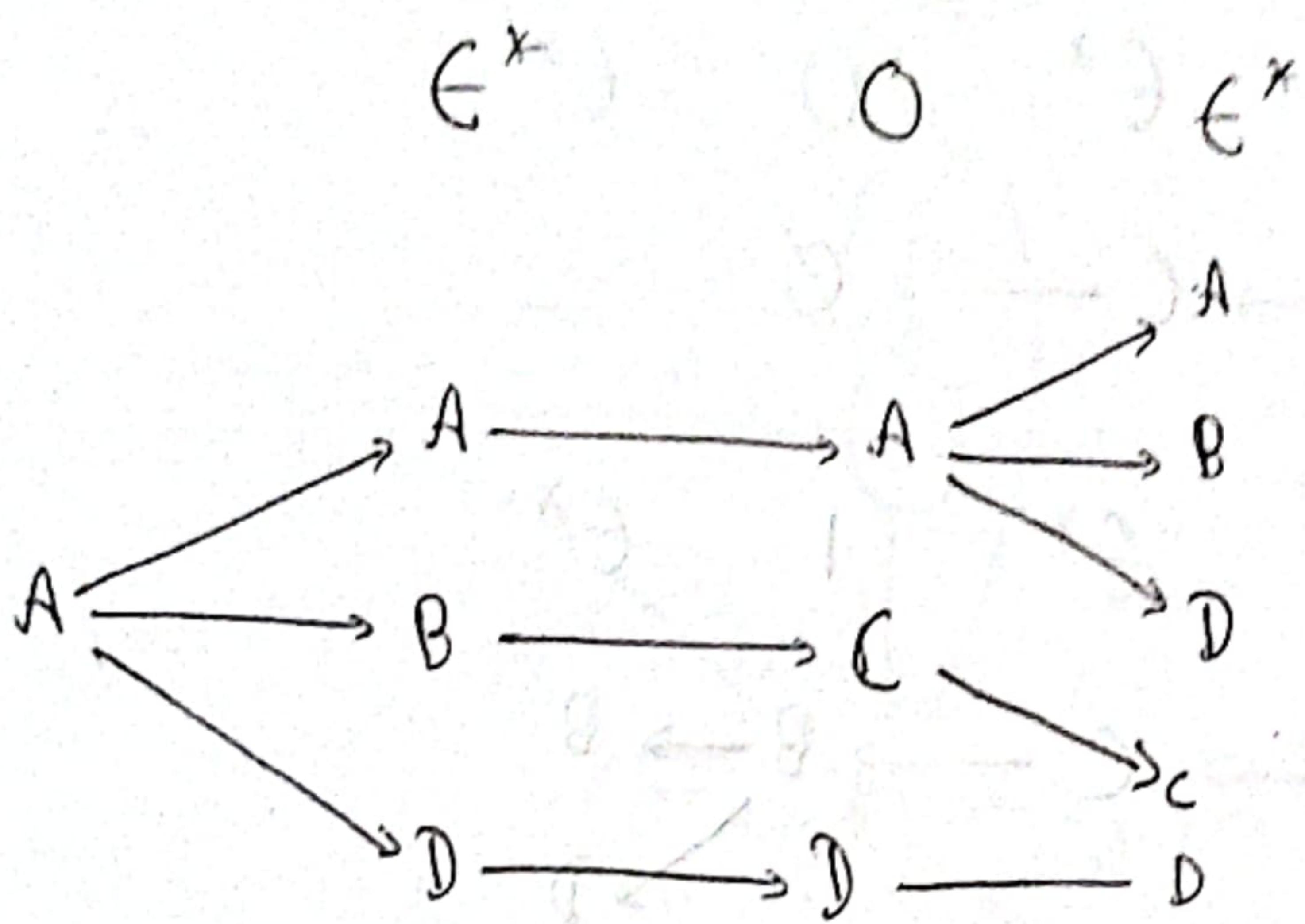
$\epsilon\text{-NFA} \rightarrow$ Without reading any character it will go to next state.

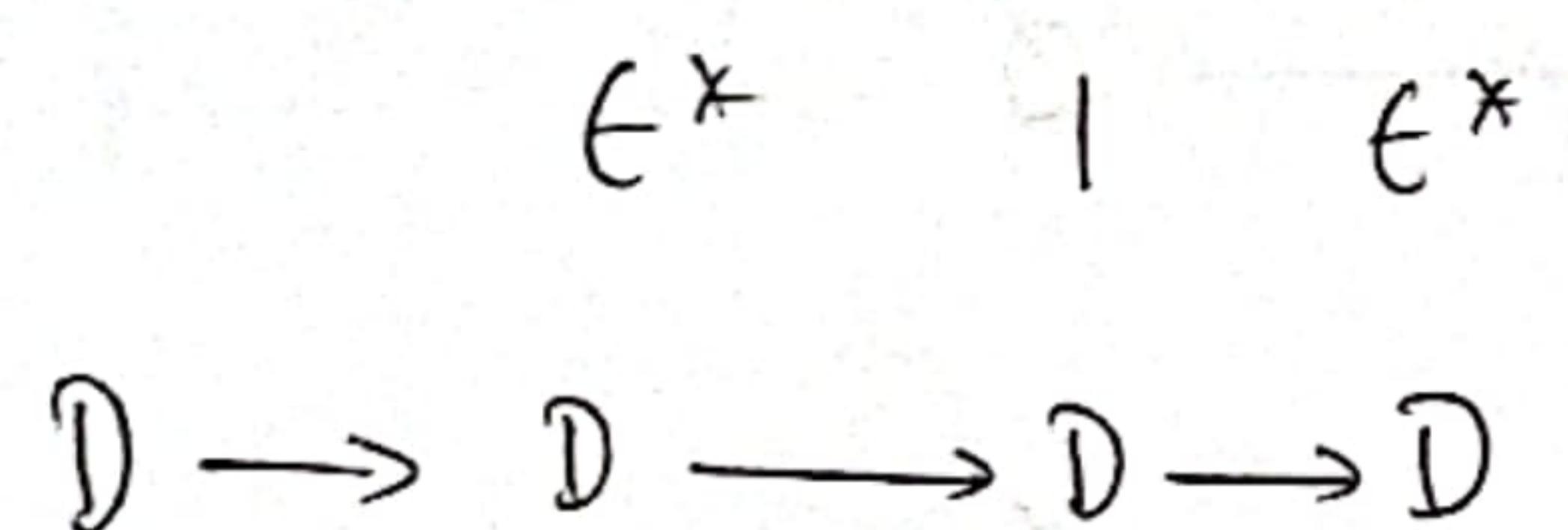
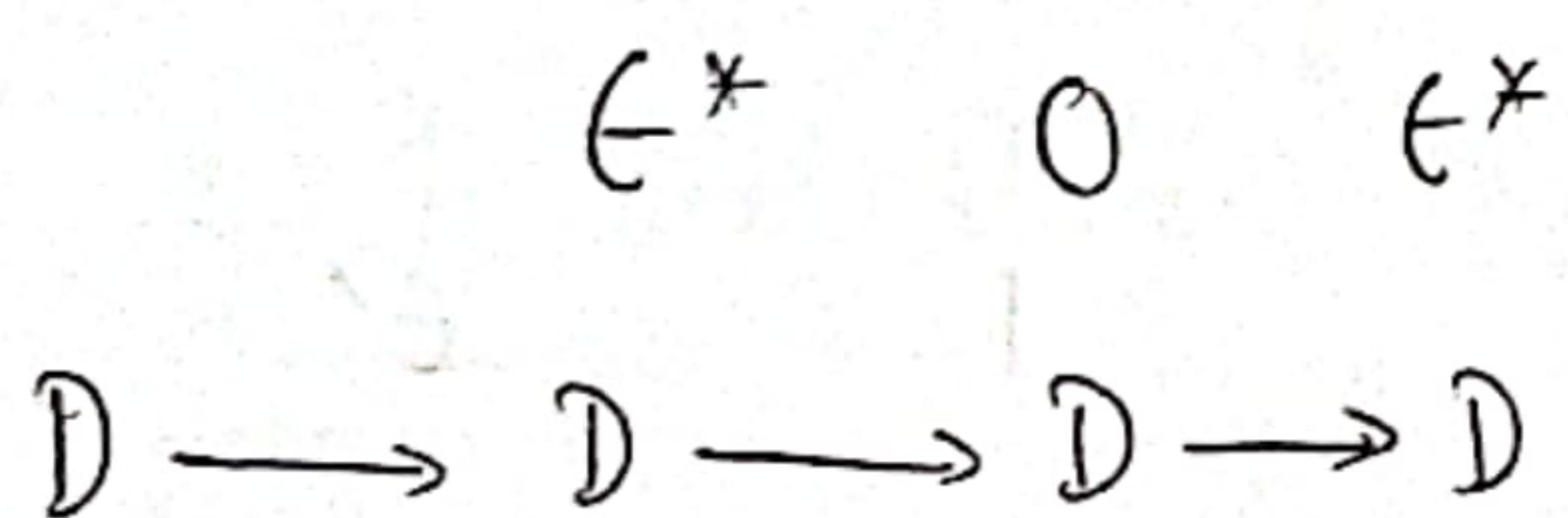
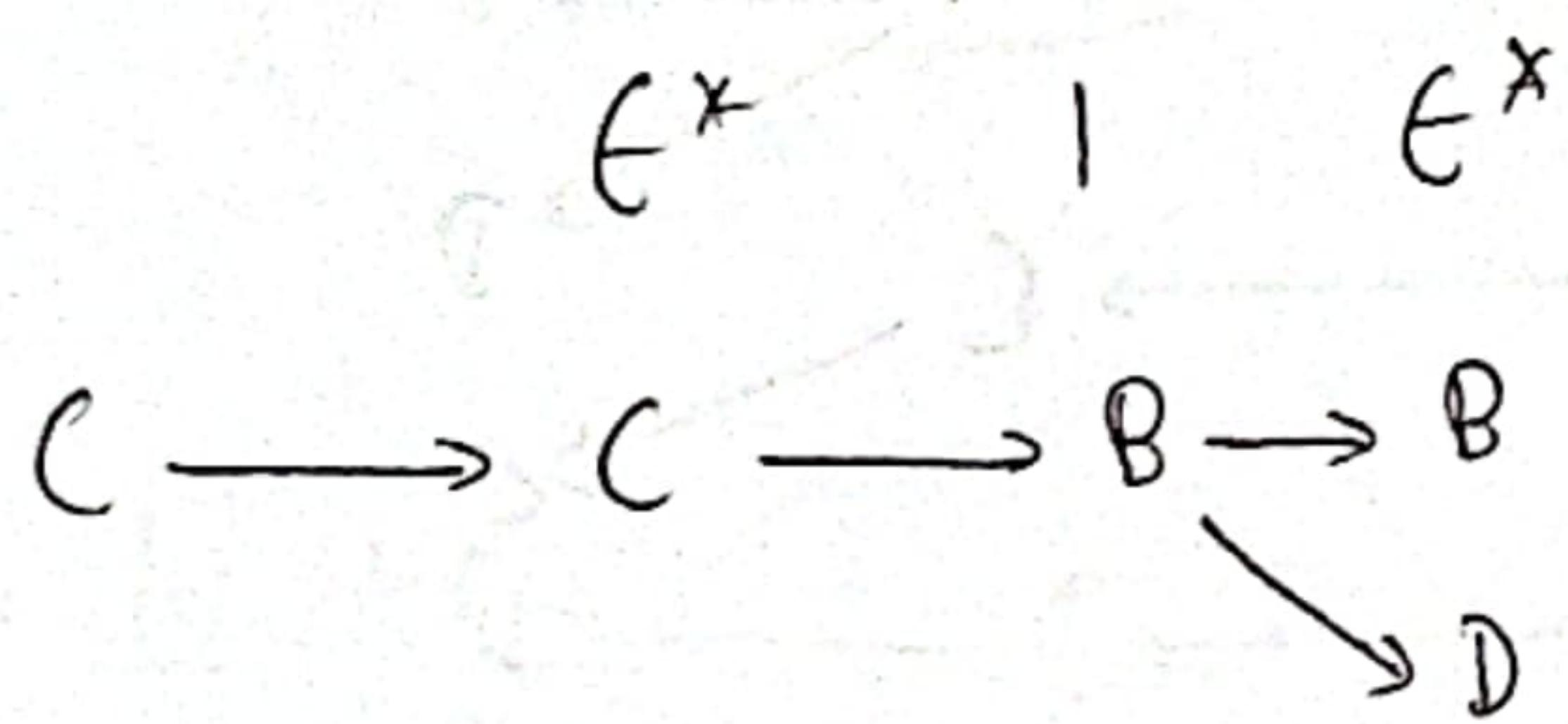
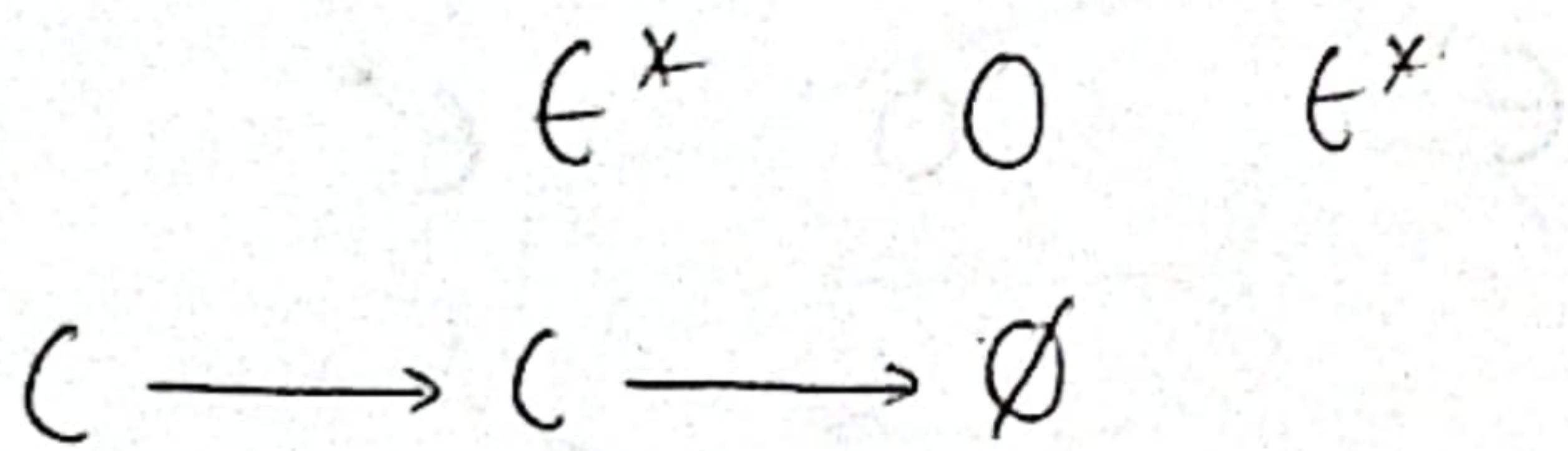
① Convert from $\epsilon\text{-NFA}$ to NFA :-



ϵ -Closure (A) :- what are states that you can reach by reading ' ϵ ' from state 'A'.

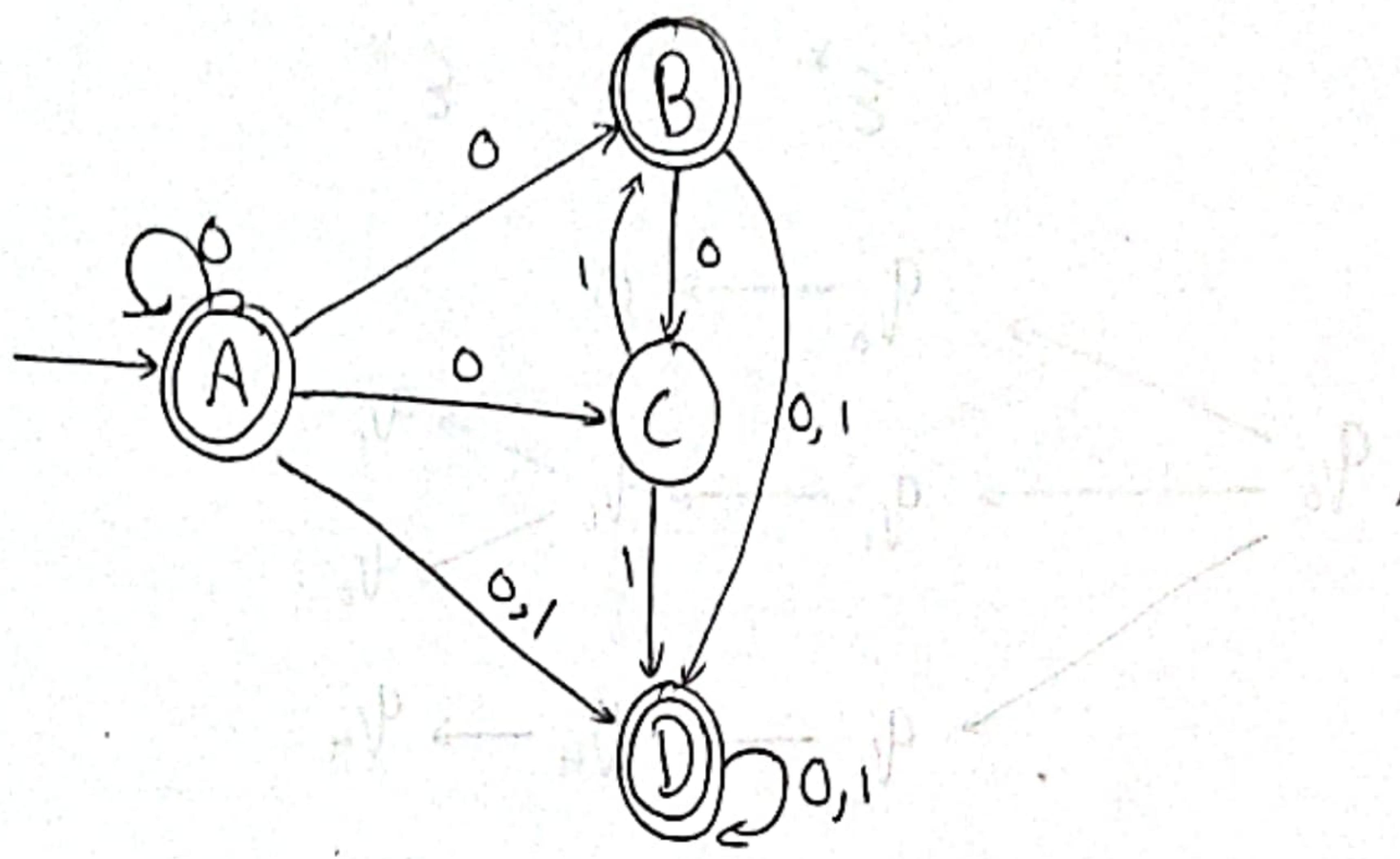
$$\Rightarrow \epsilon\text{-Closure (A)} = \{A, B, D\}$$



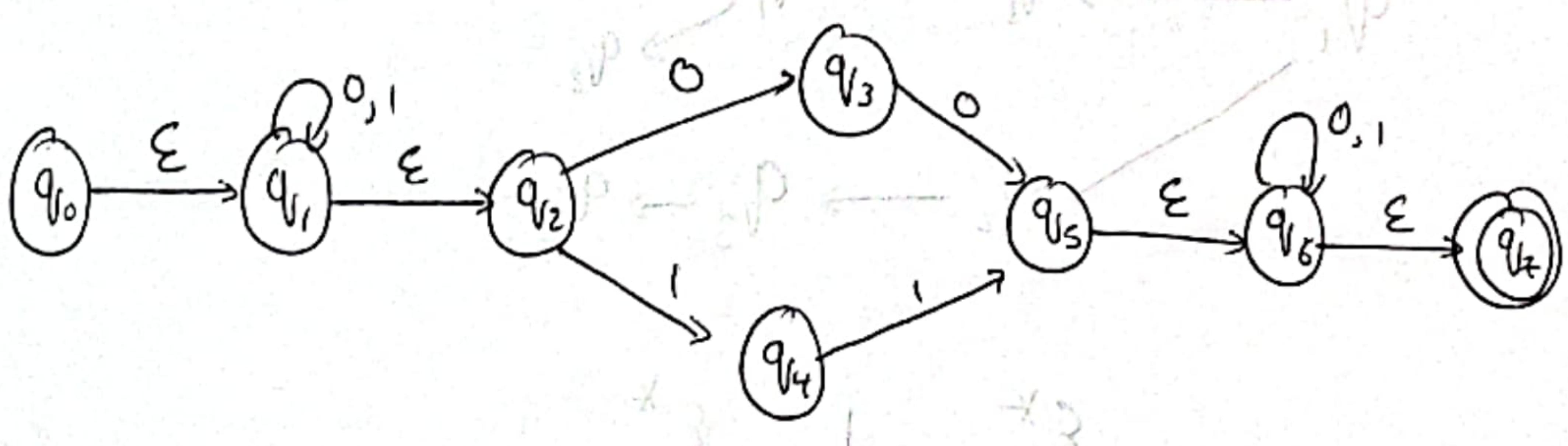


	0	1
$\rightarrow *A$	$\{A, B, C, D\}$	$\{D\}$
B	$\{C, D\}$	$\{D\}$
$*C$	$\{\emptyset\}$	$\{B, D\}$
$*D$	$\{D\}$	$\{D\}$

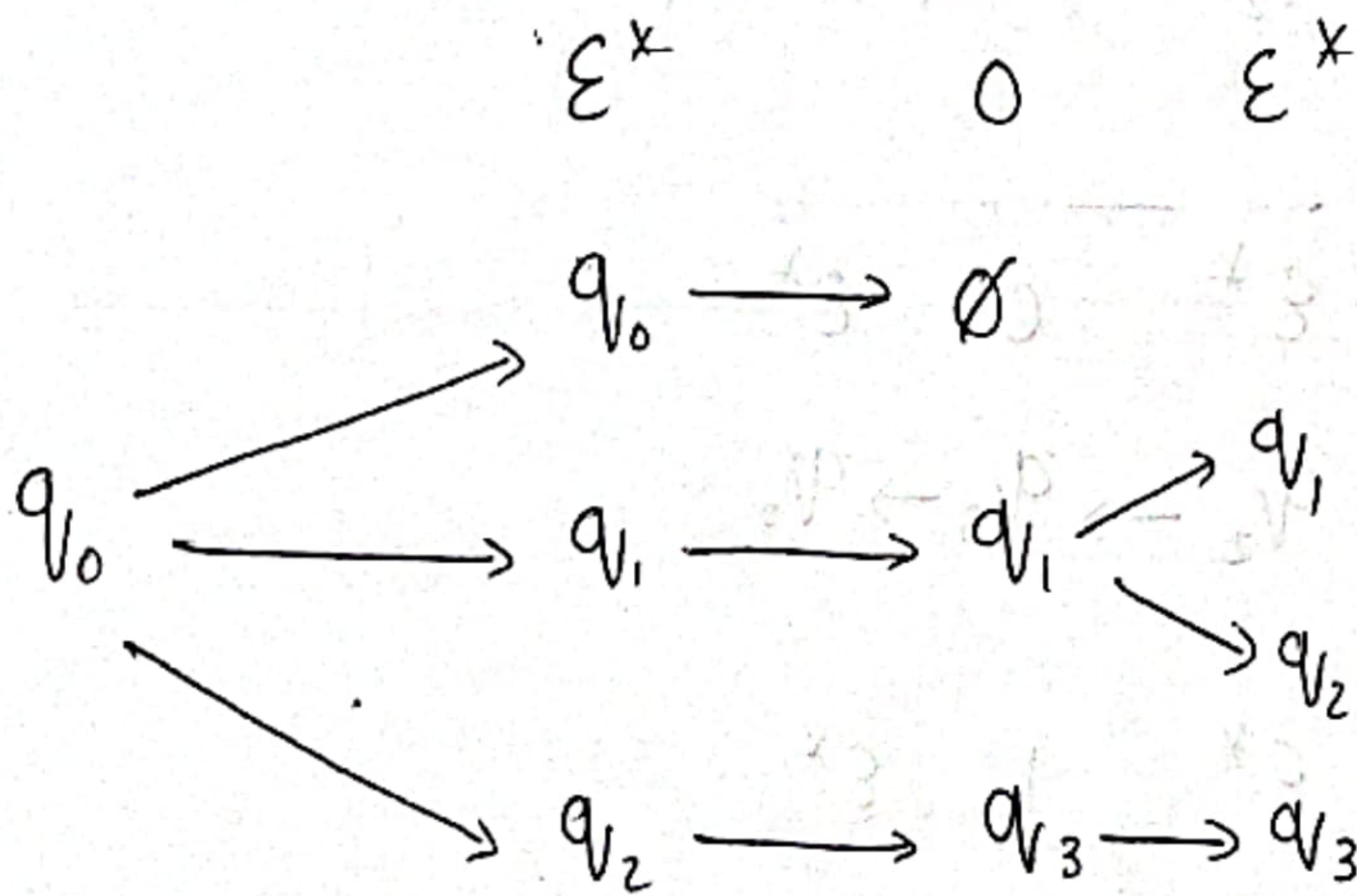
Final States:- The states that can be reached from start reading t

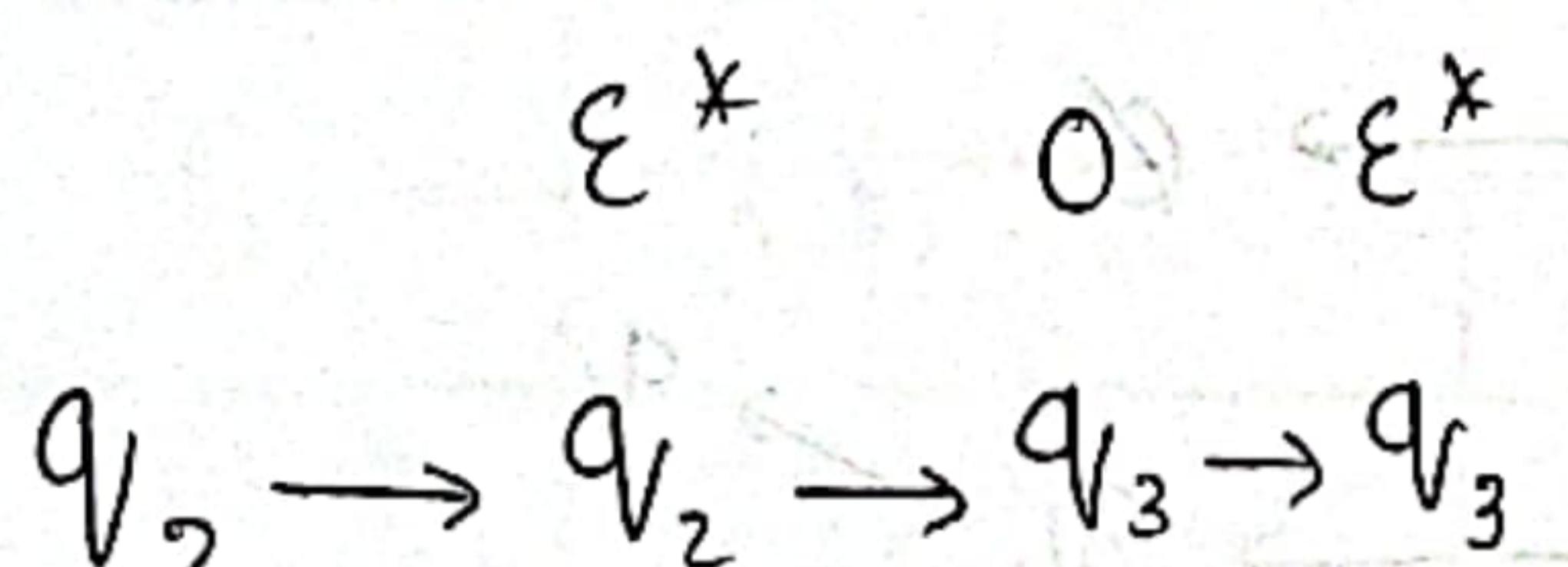
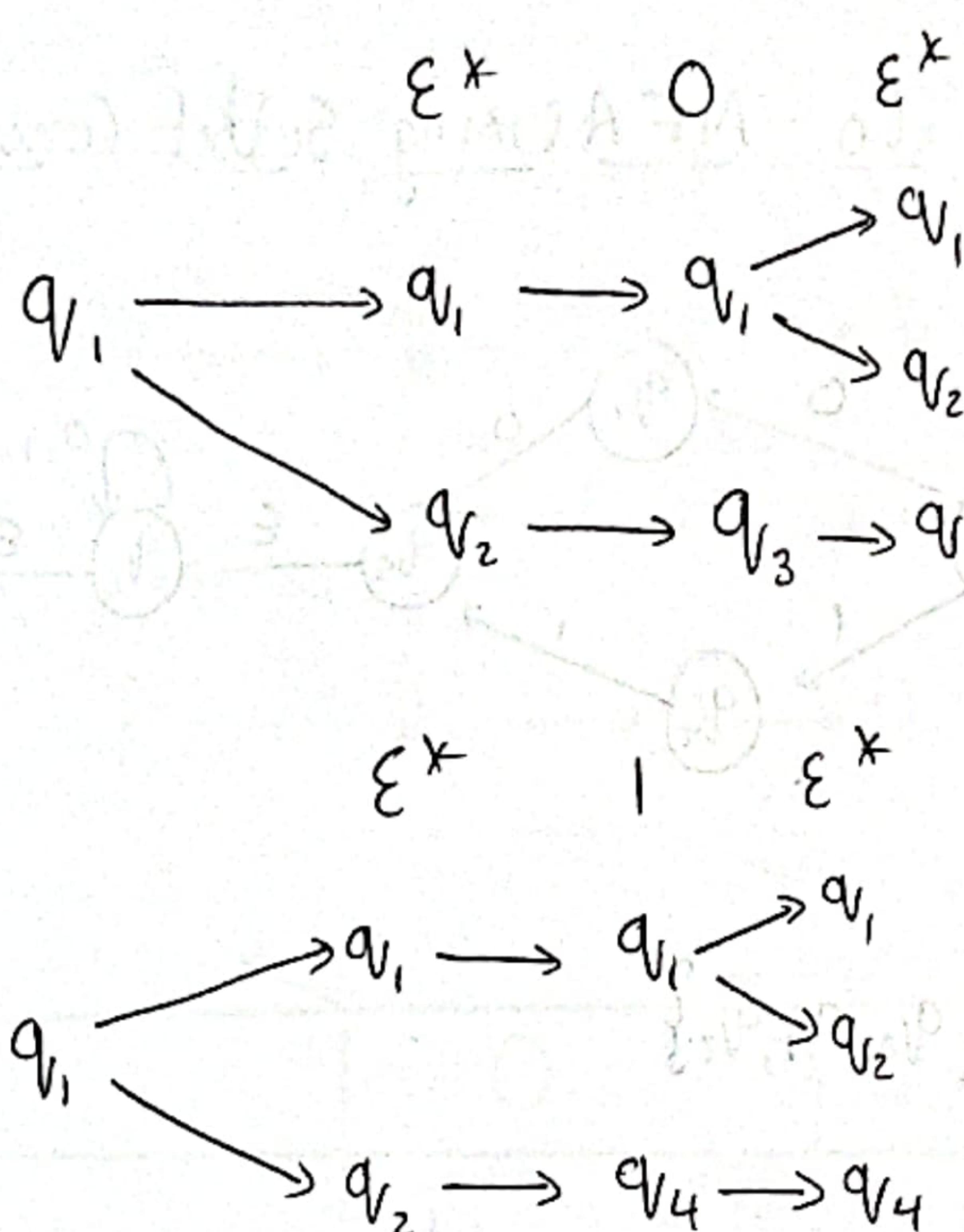
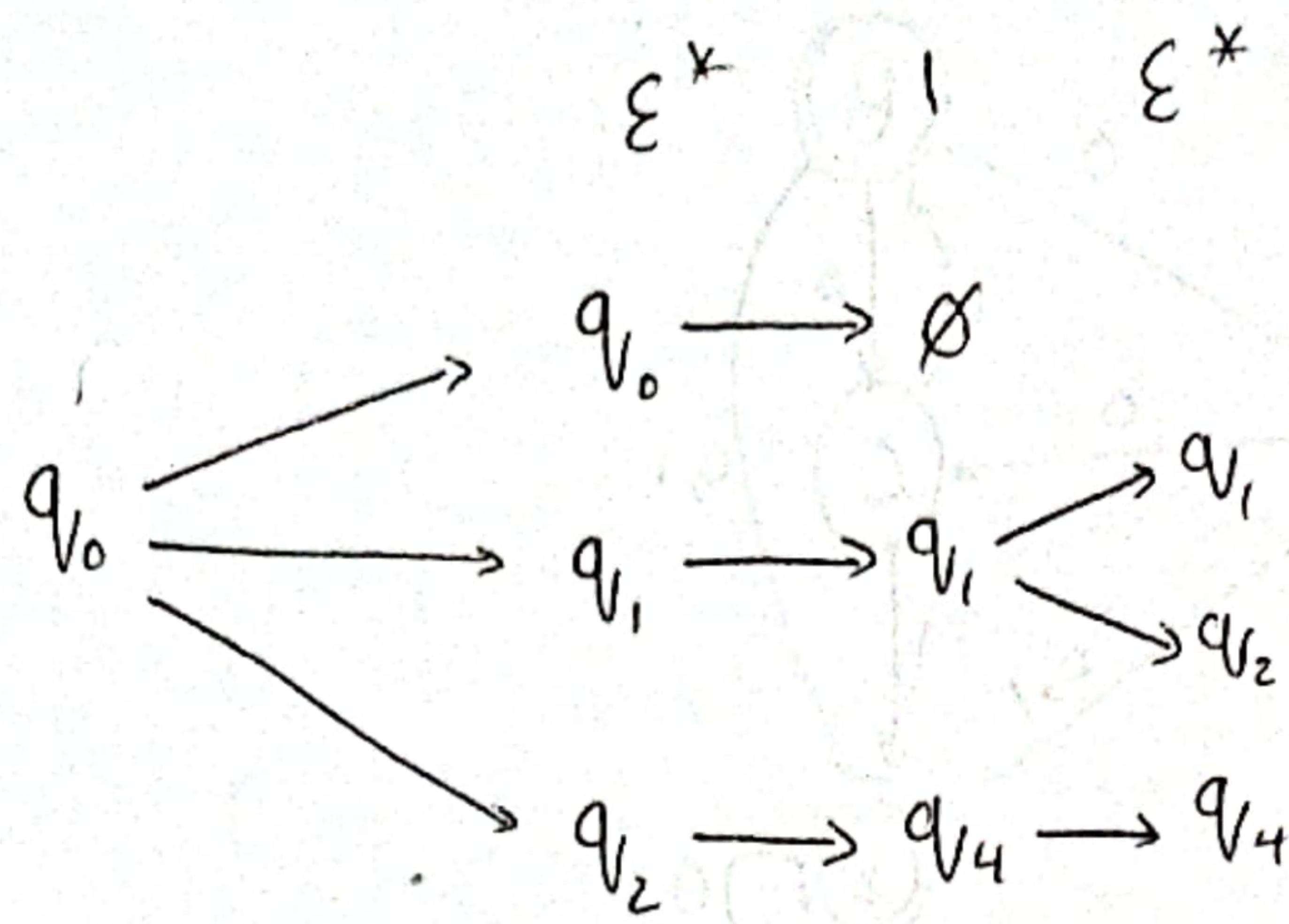


② Convert ϵ -NFA to NFA using Subset Construction Method:



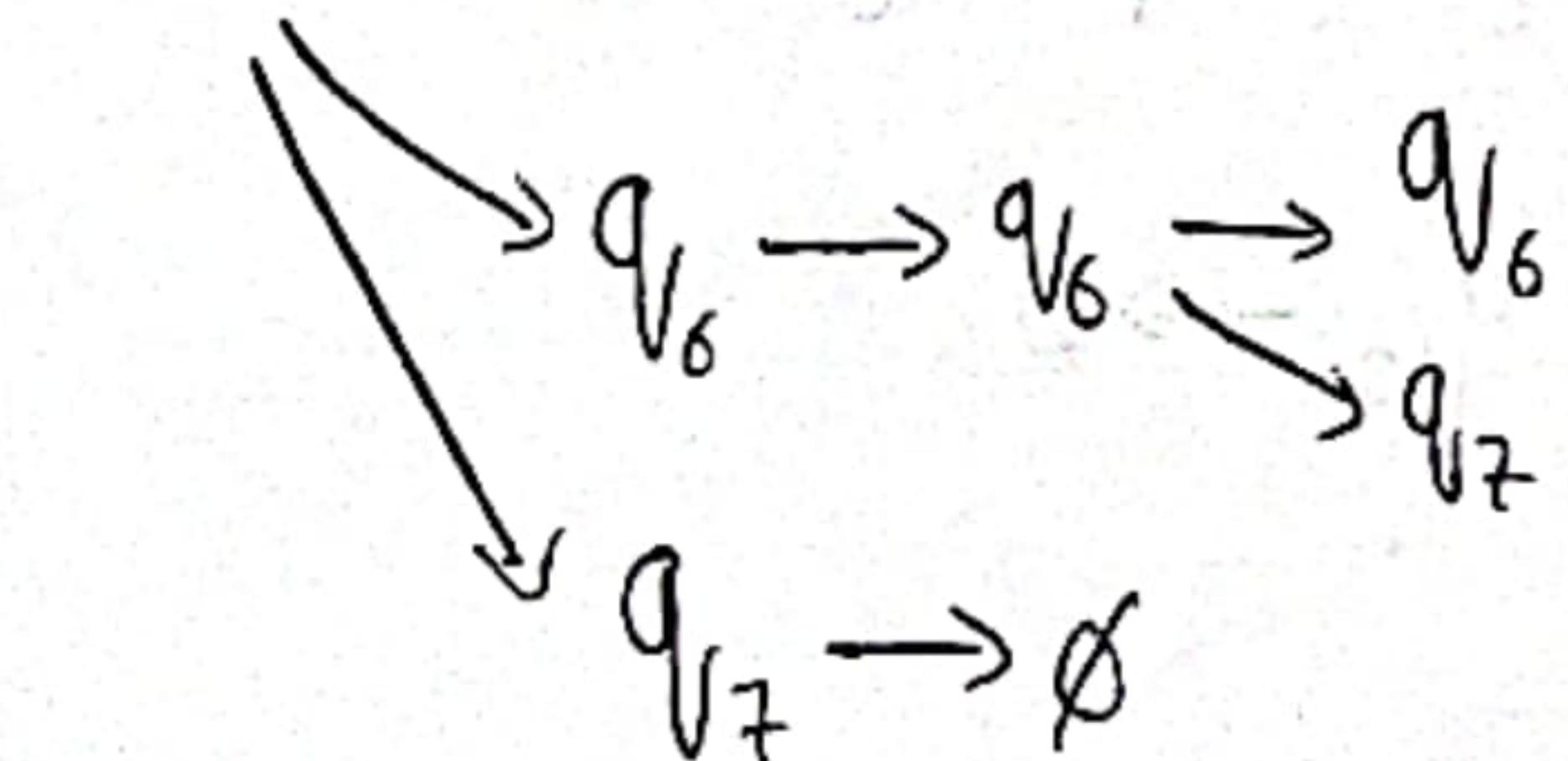
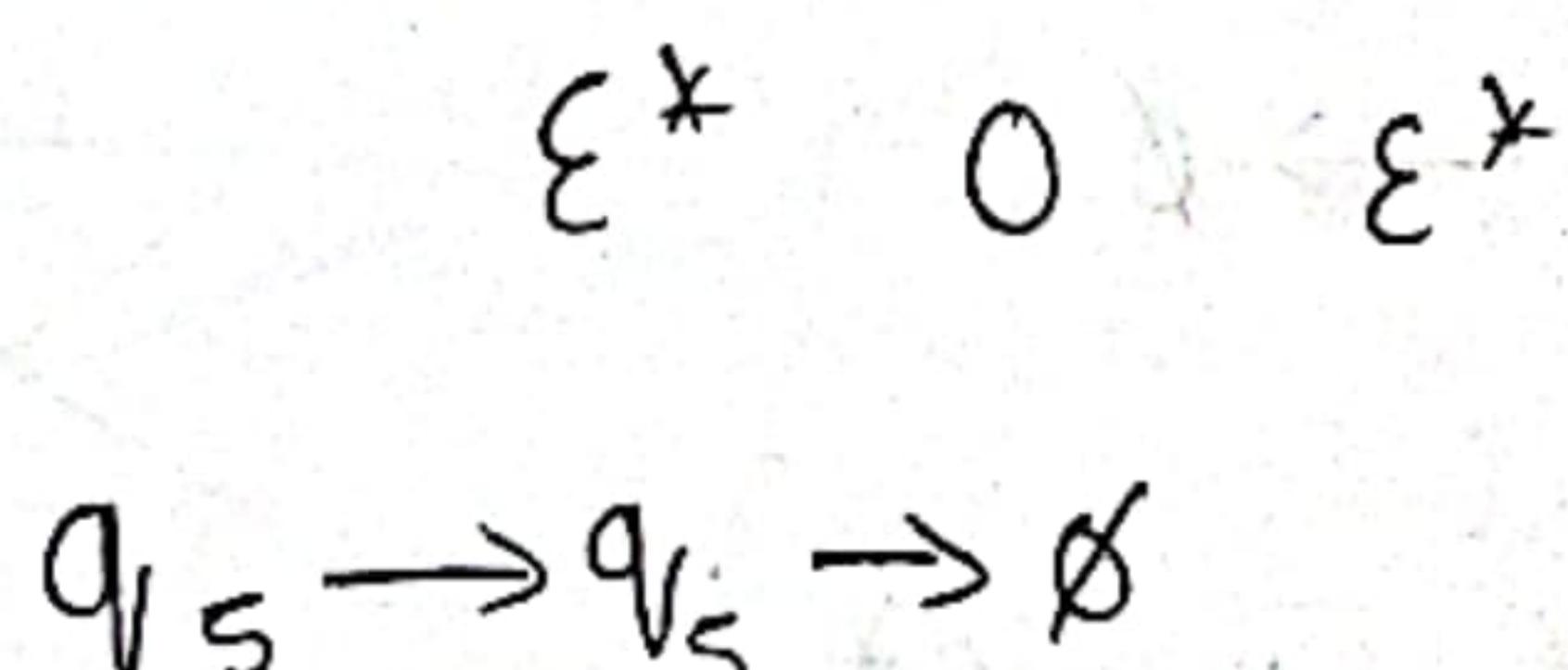
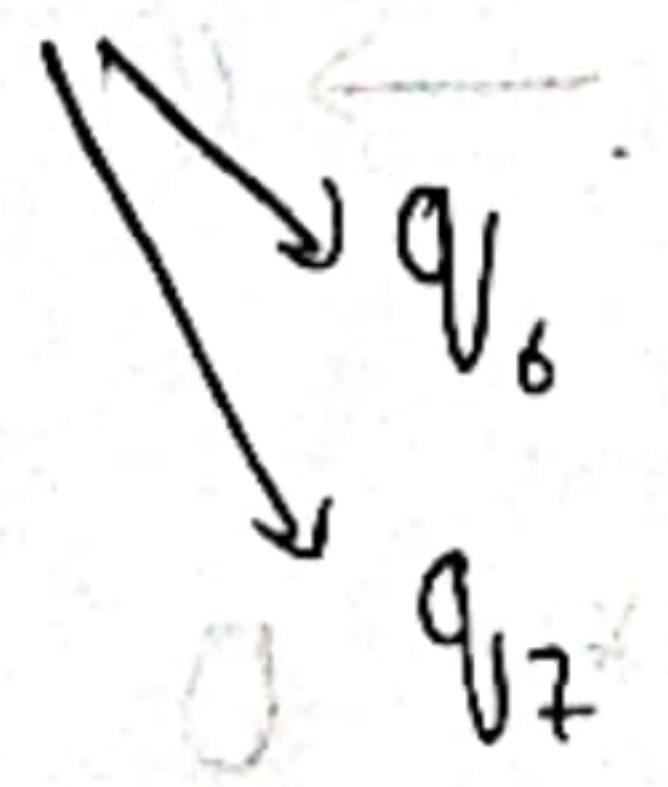
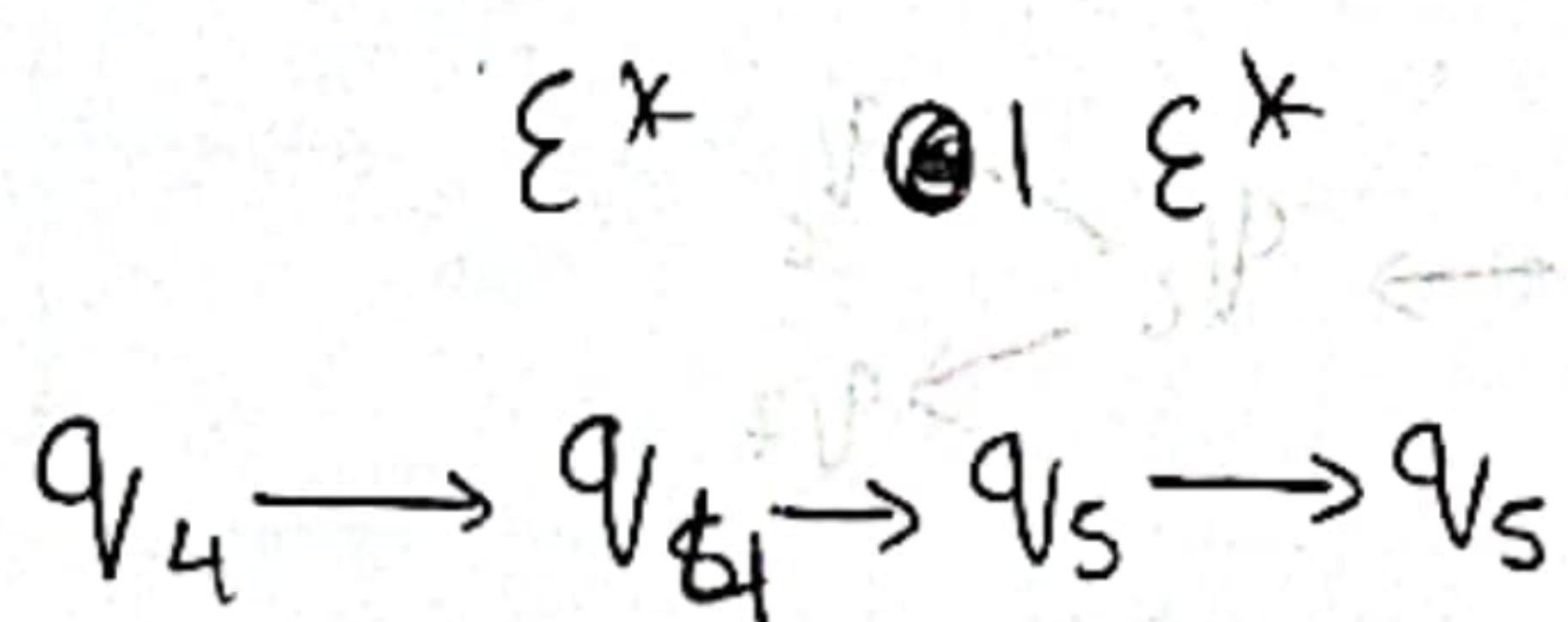
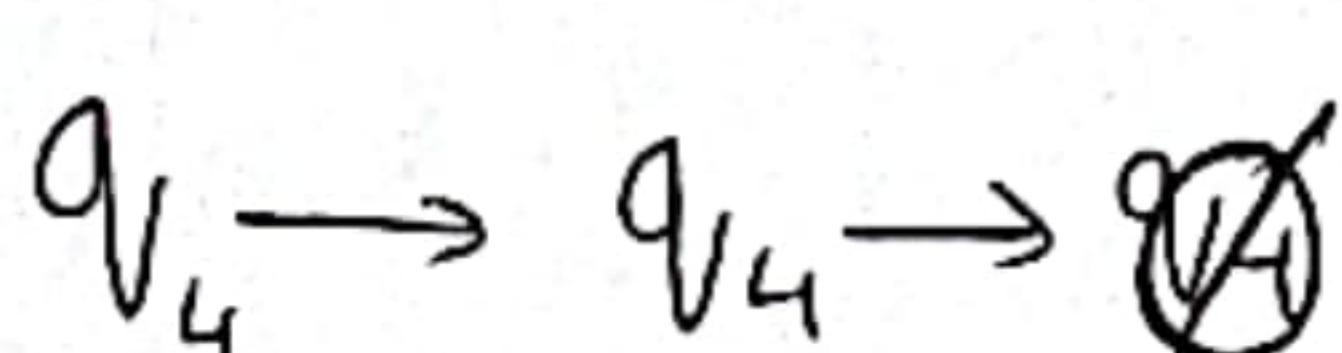
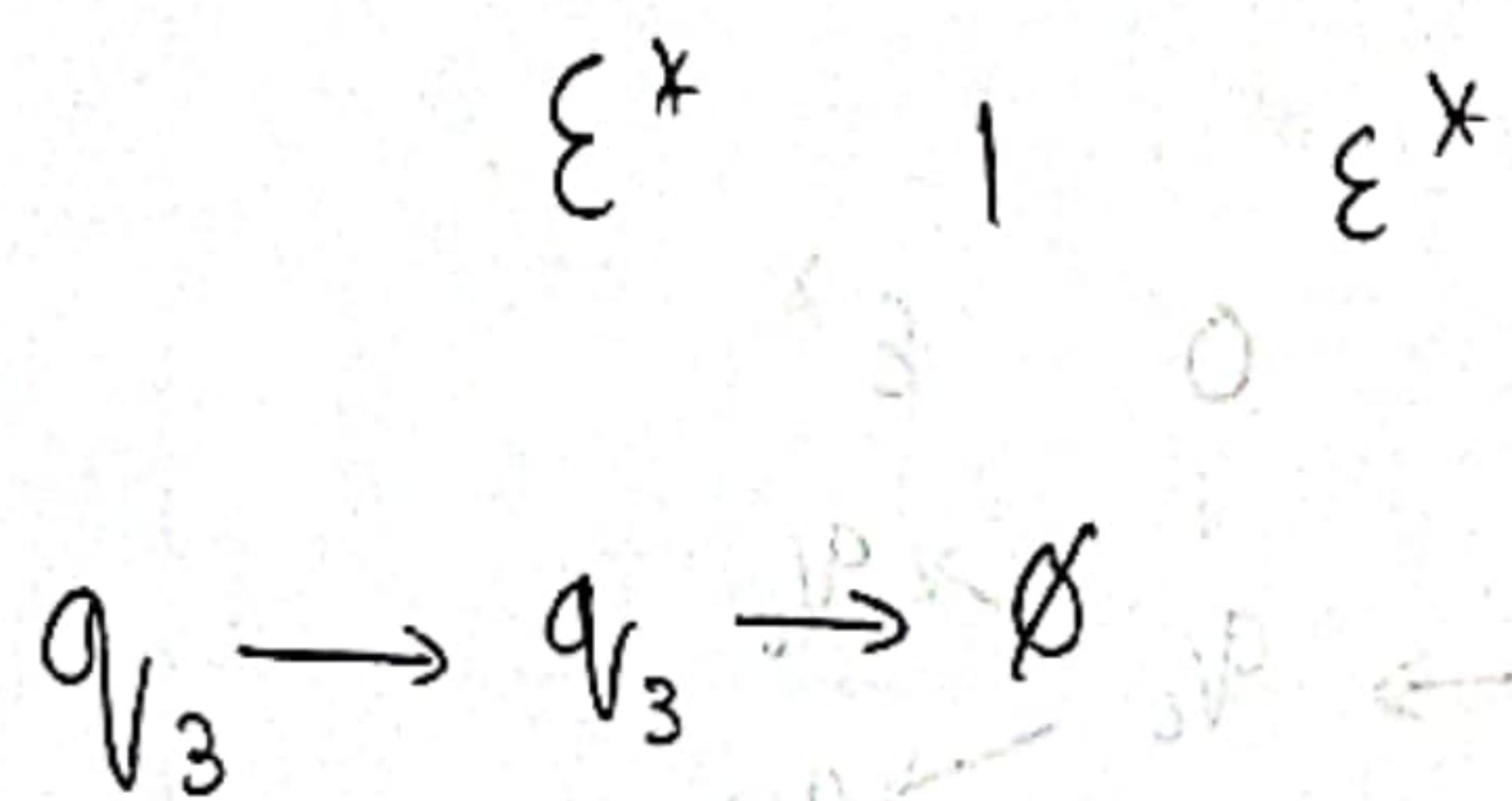
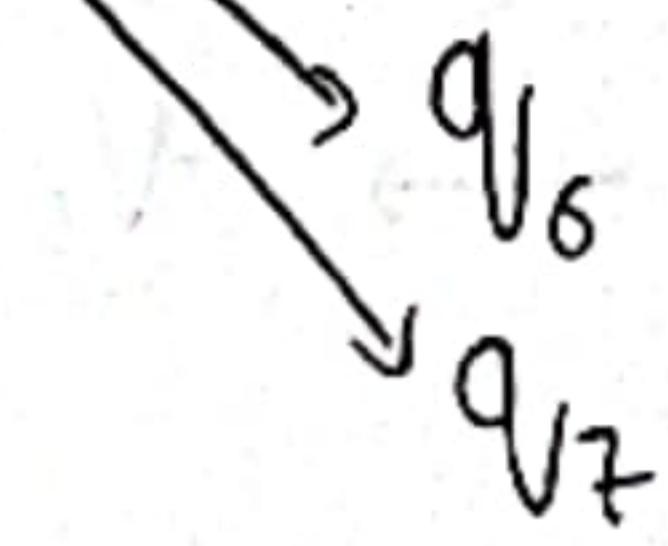
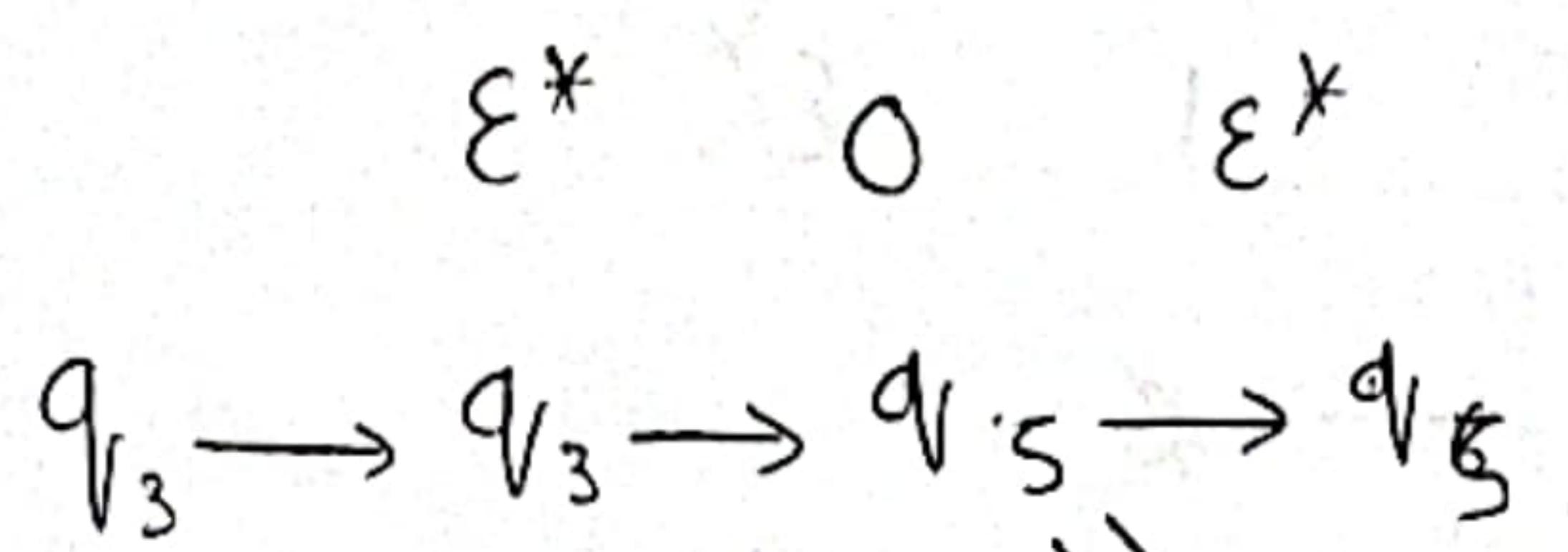
$$\epsilon\text{-closure}(q_{v_0}) = \{q_{v_0}, q_{v_1}, q_{v_2}\}$$

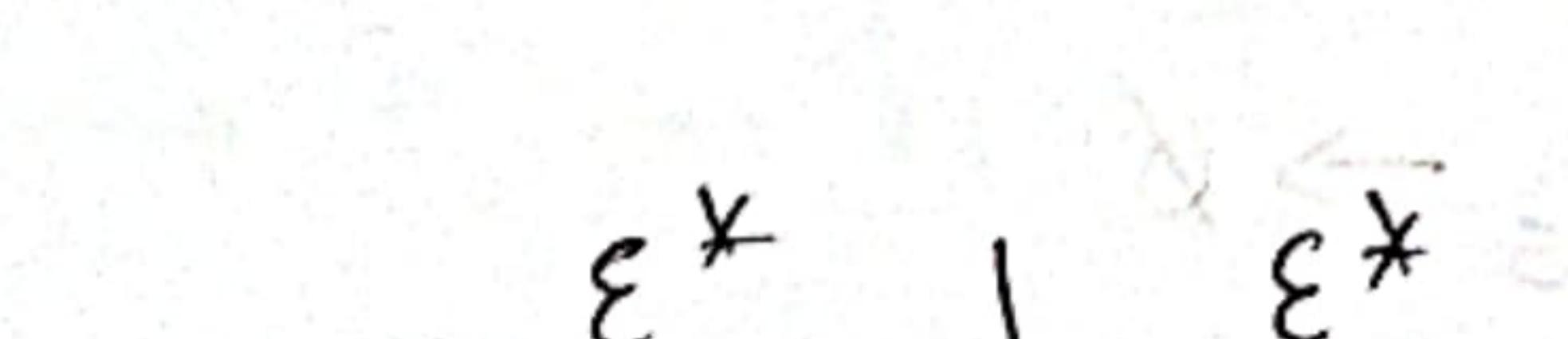
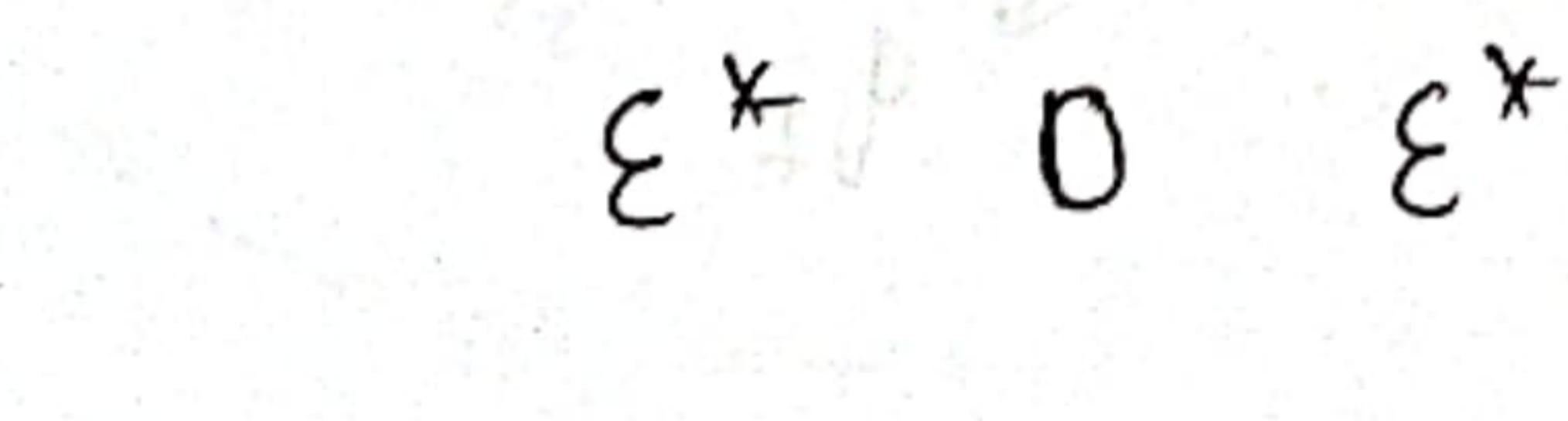
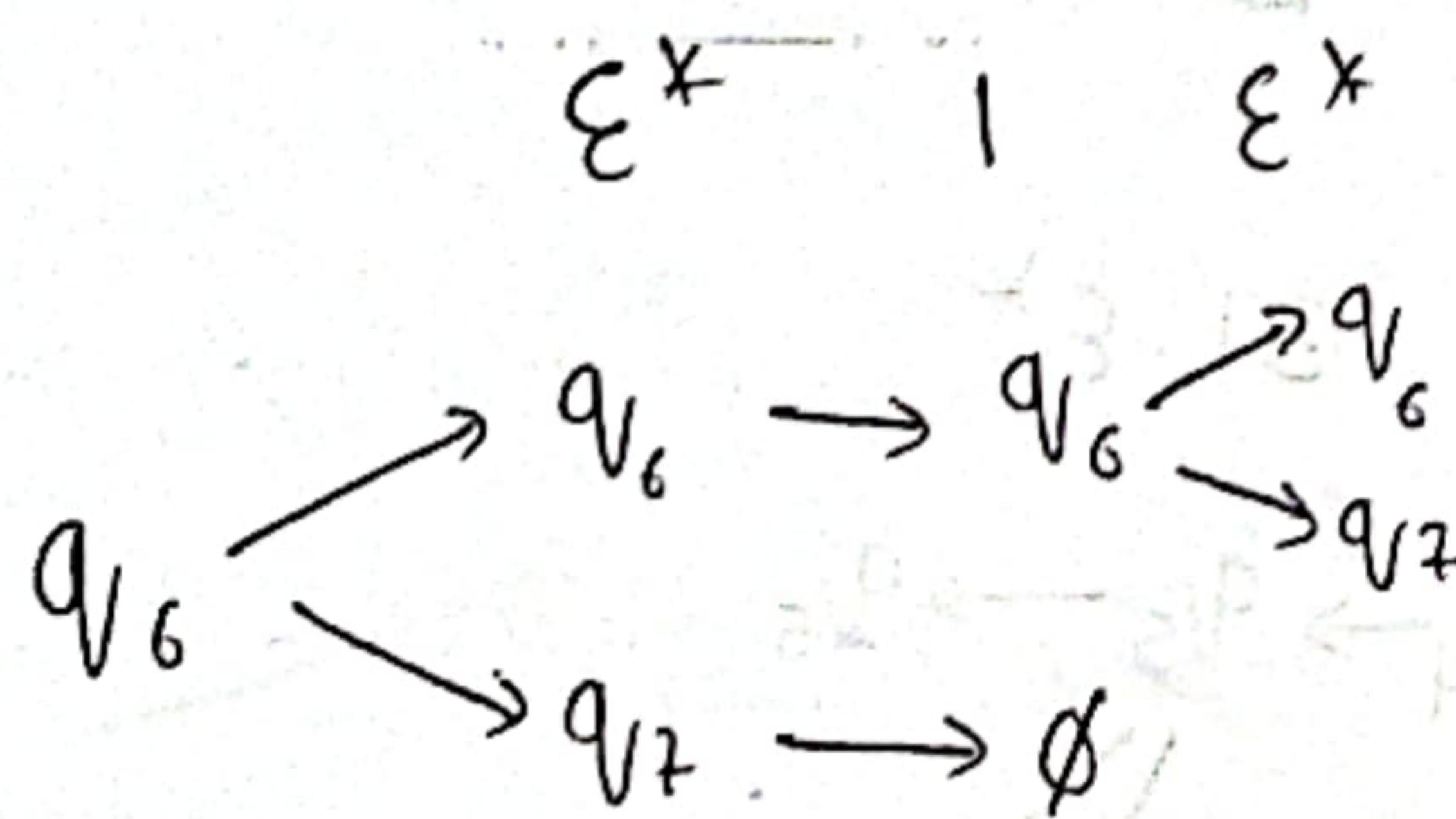
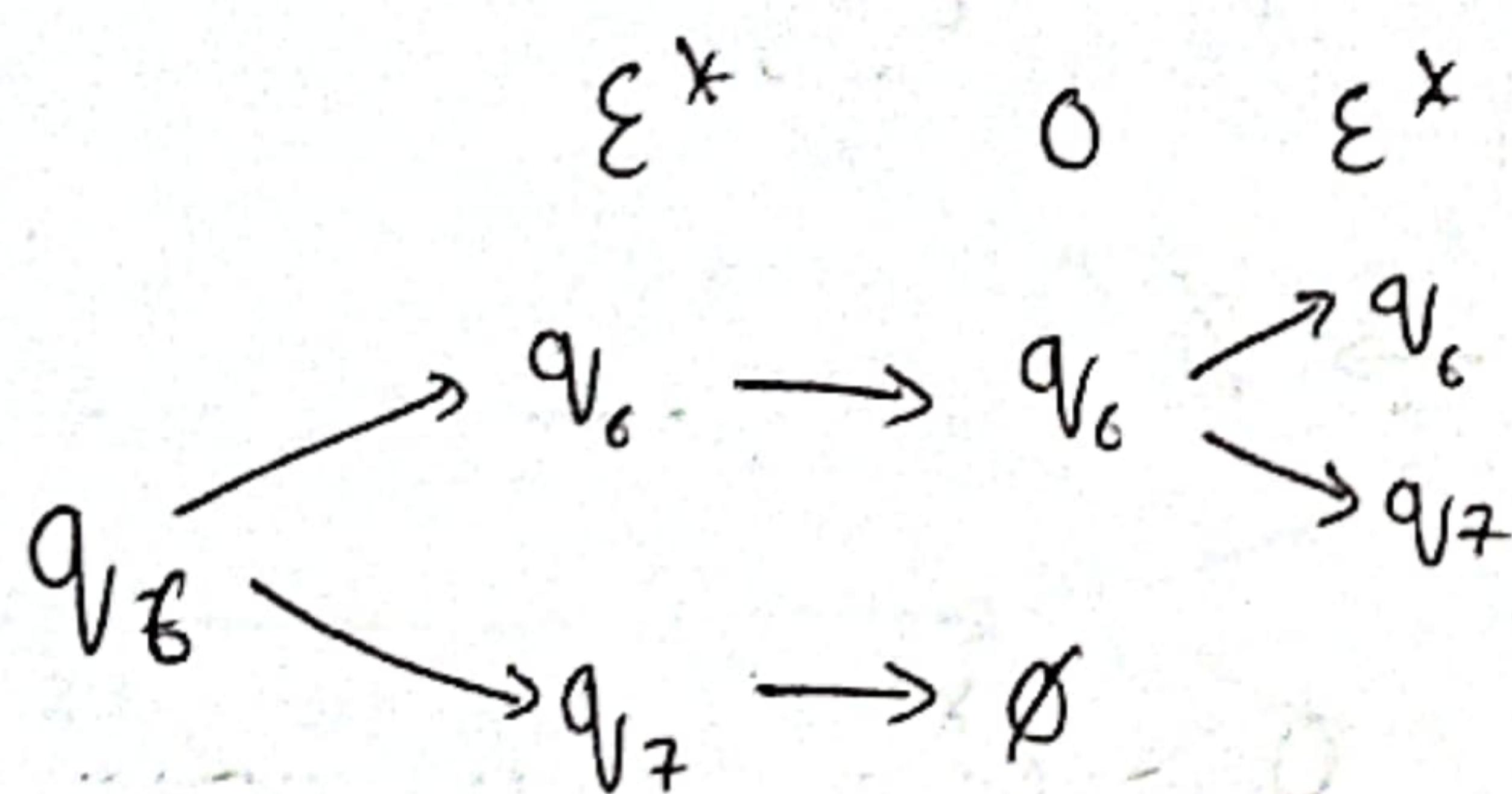
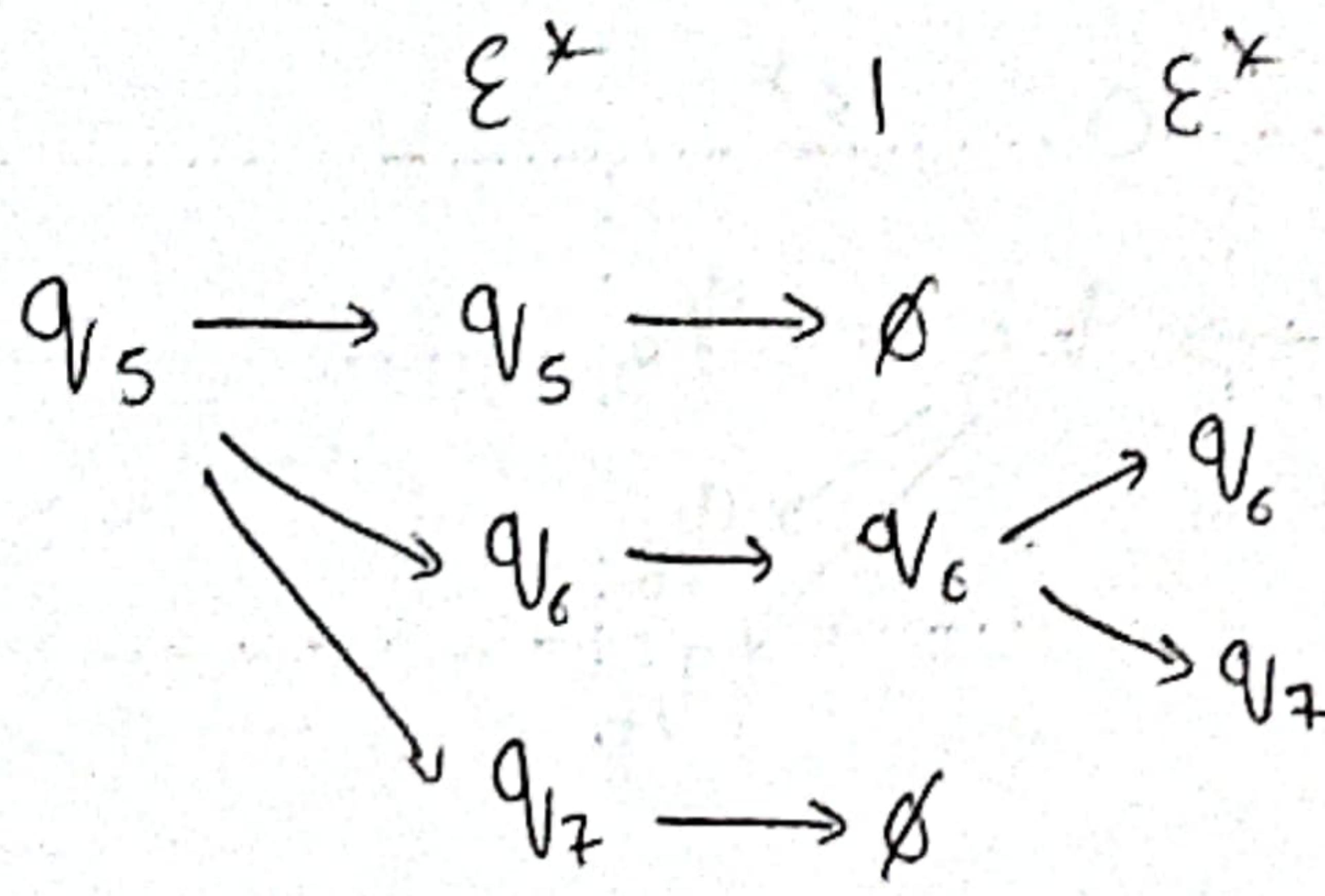




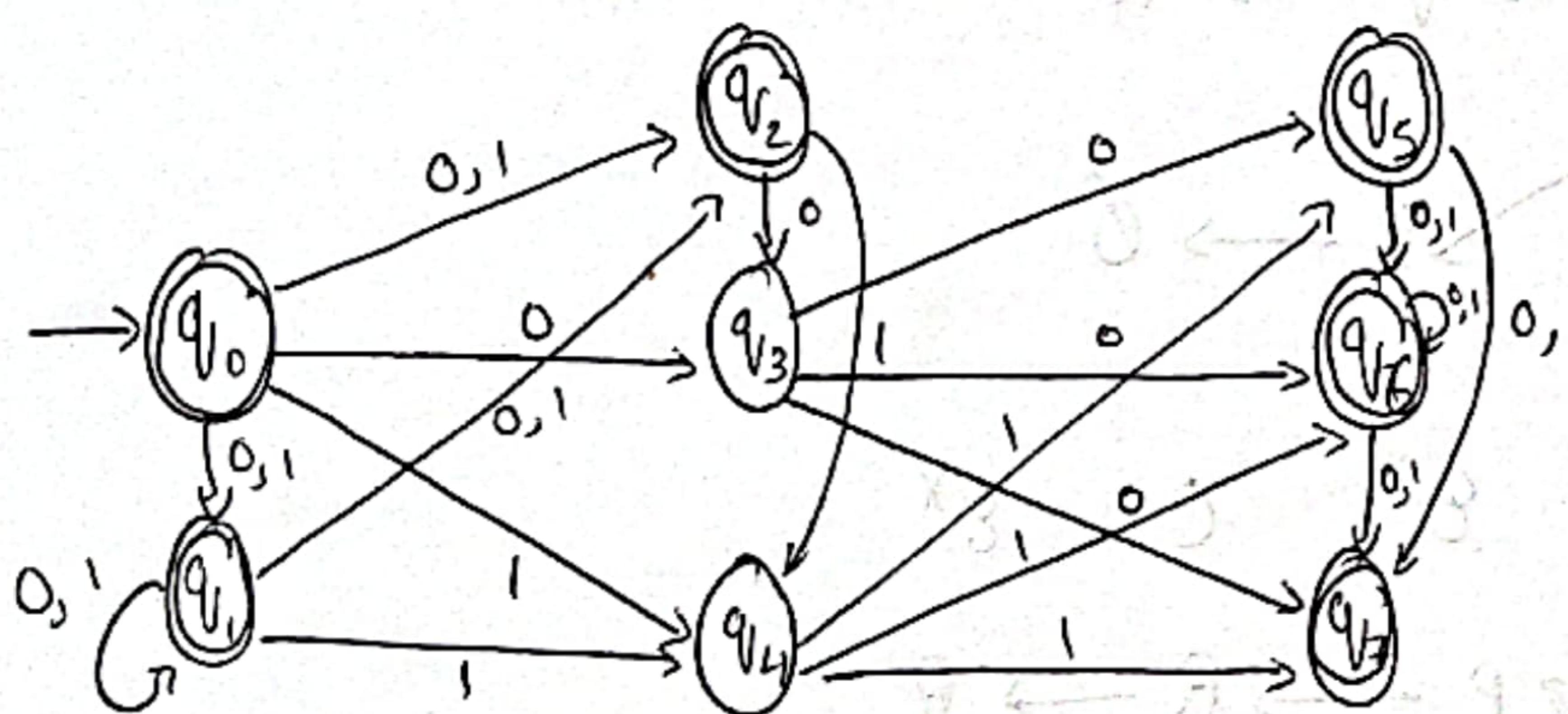
$\emptyset \quad \varepsilon^* \quad 1 \quad \varepsilon^*$

$q_2 \rightarrow q_2 \rightarrow q_4 \rightarrow q_4$

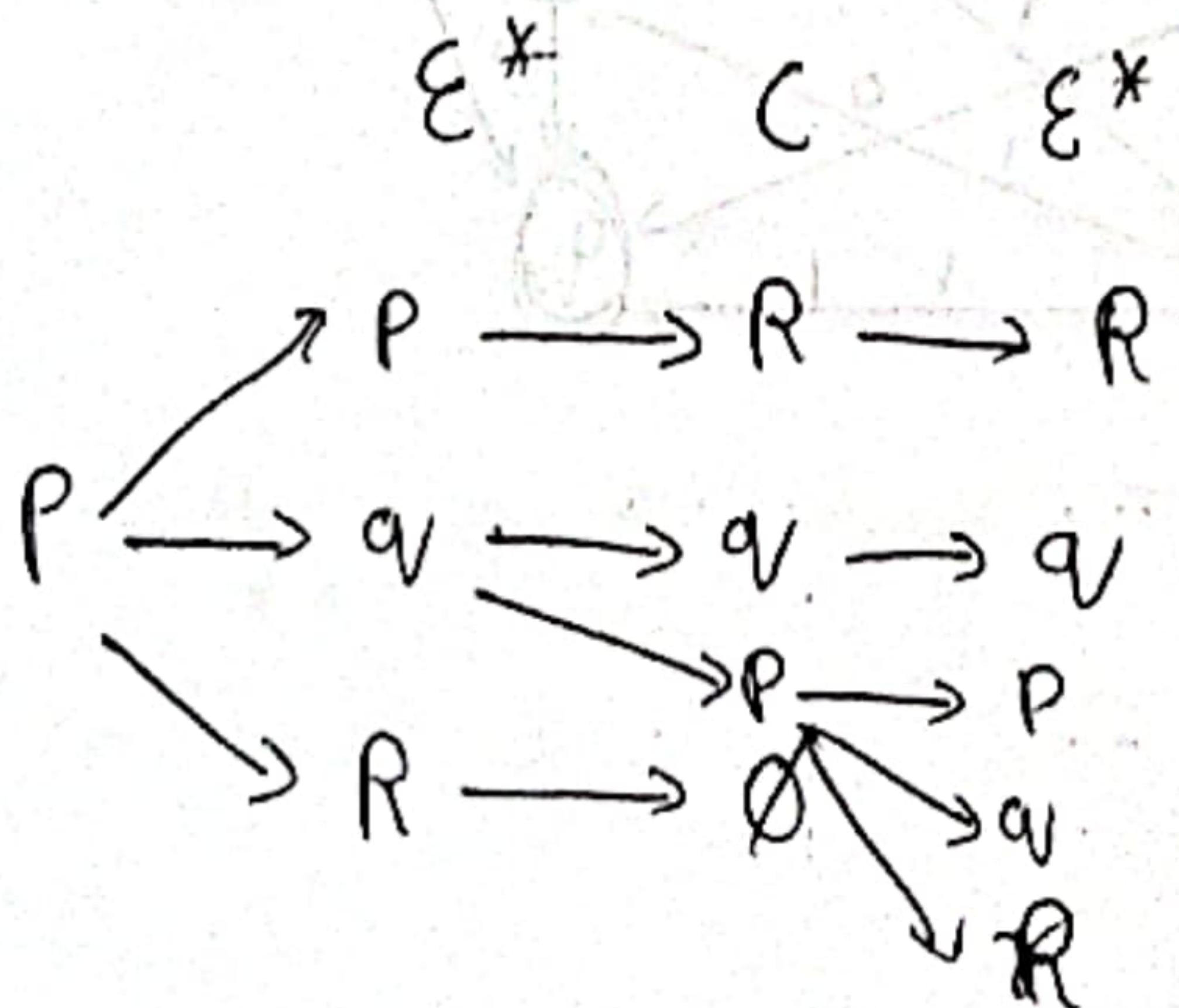
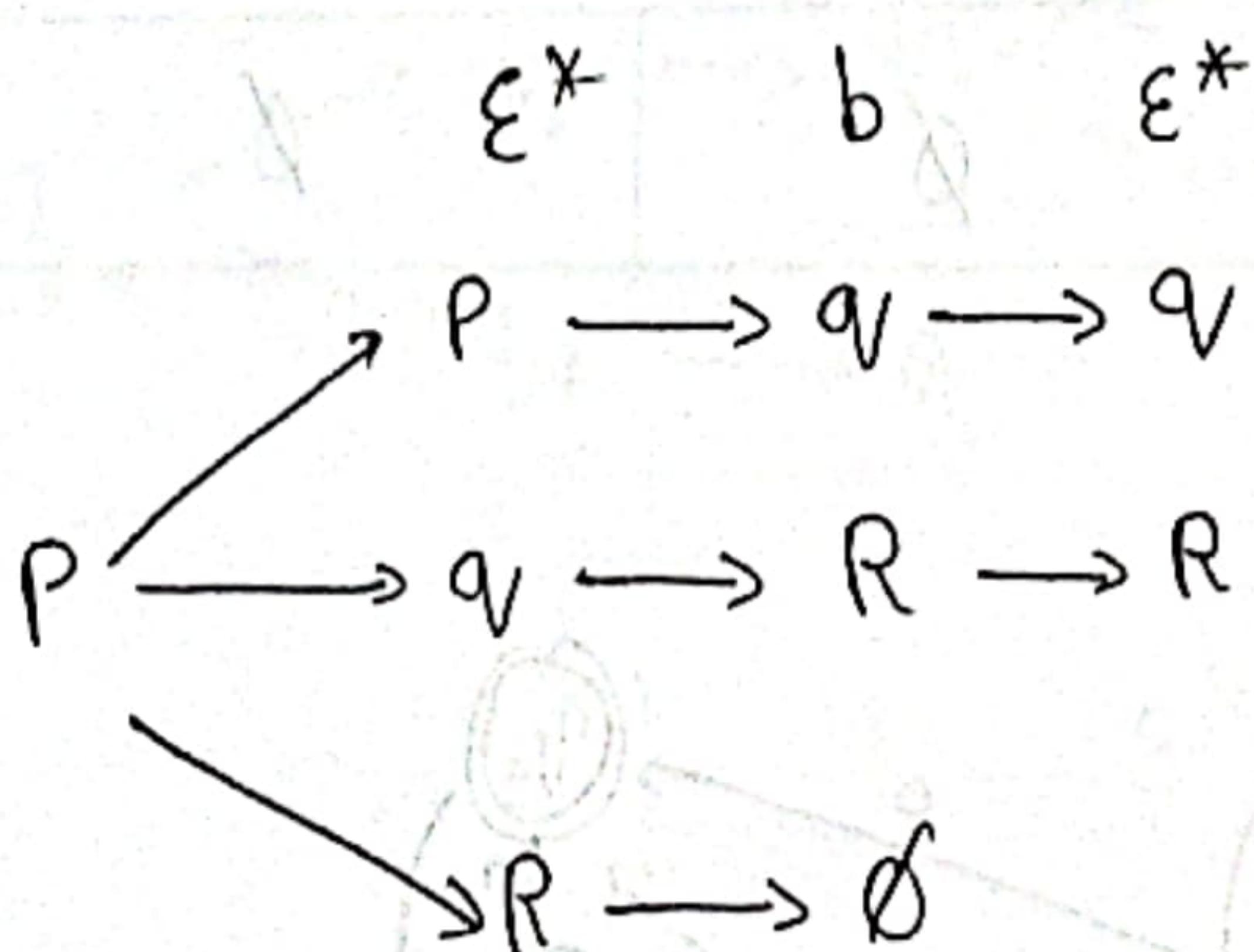
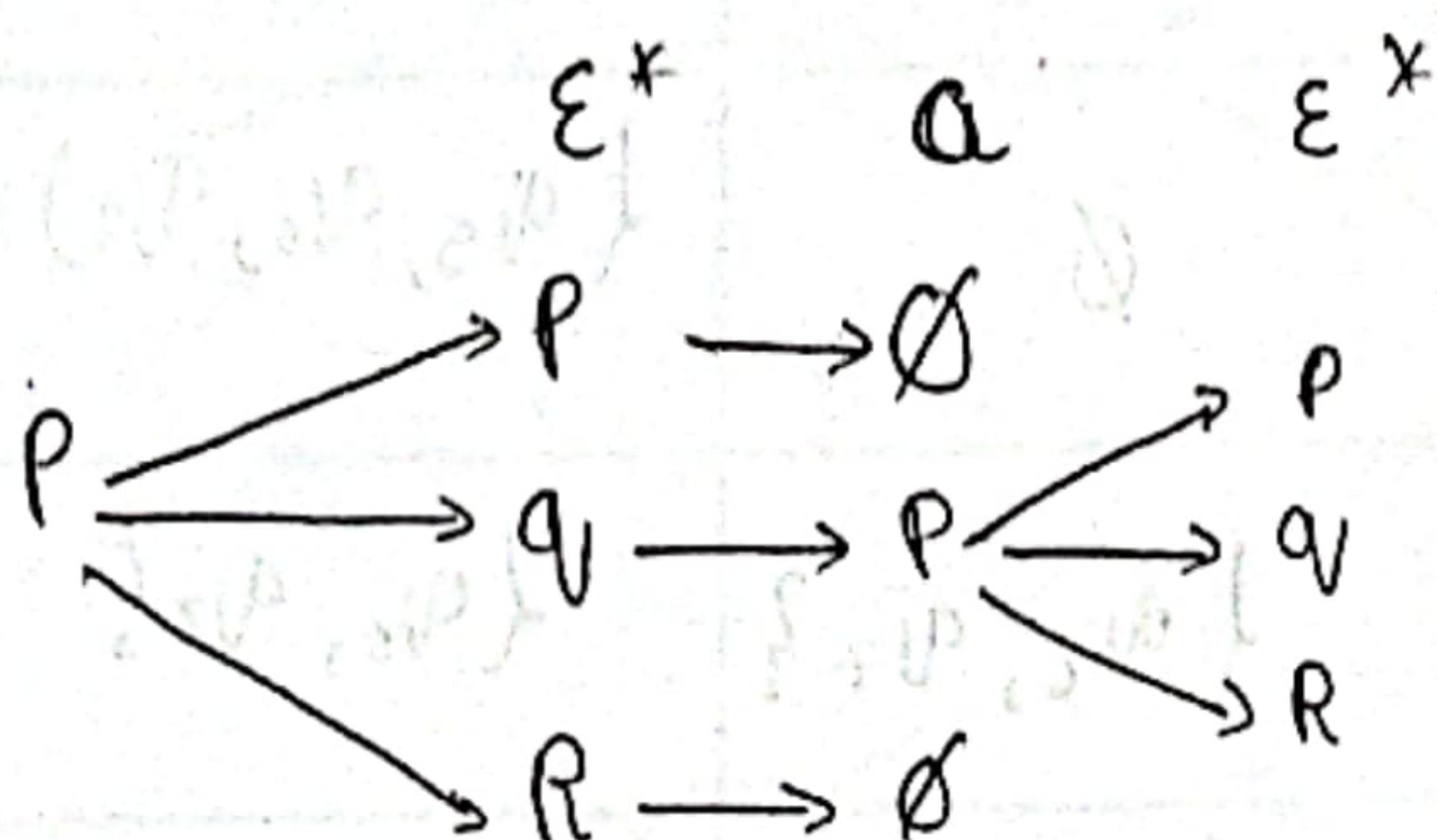
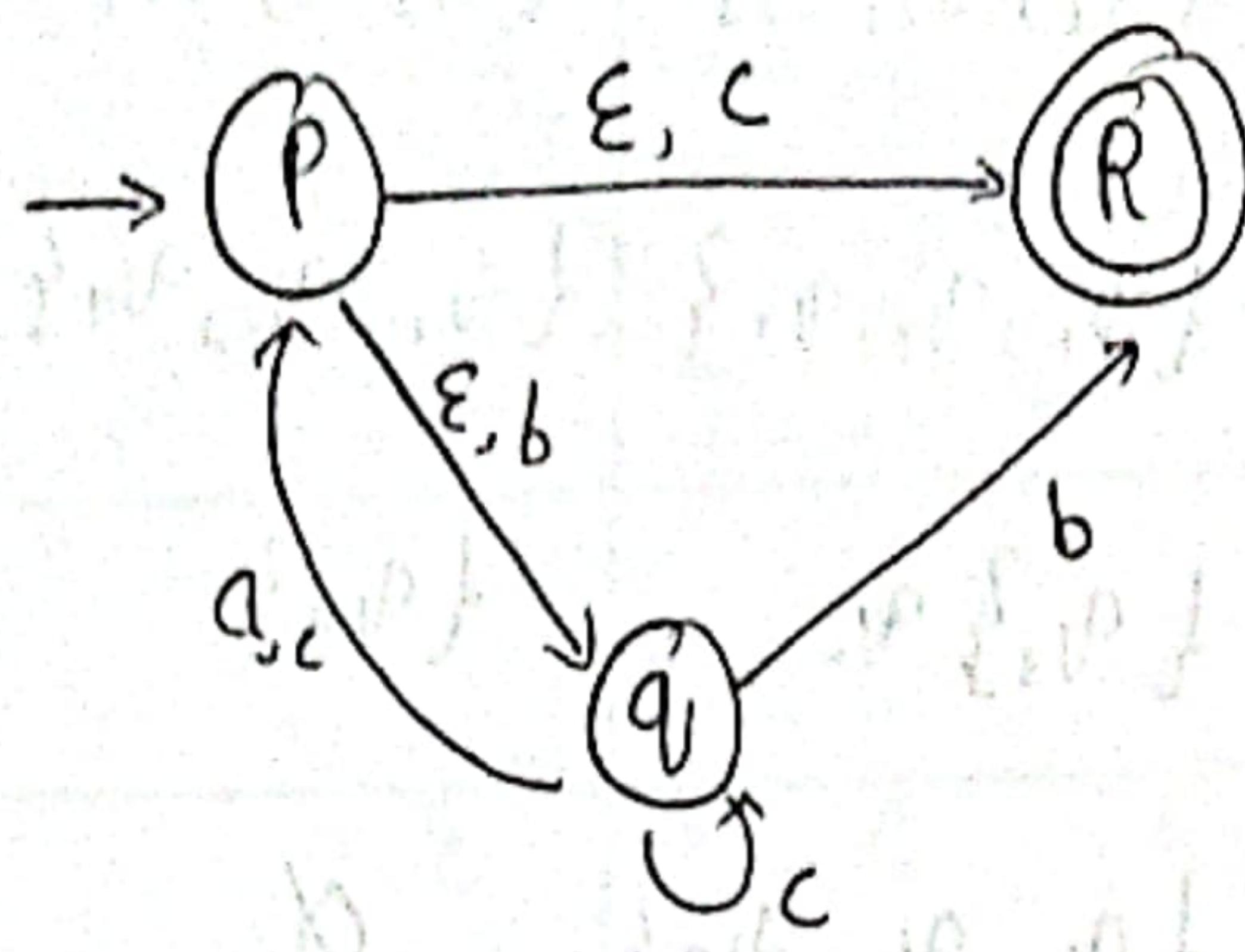


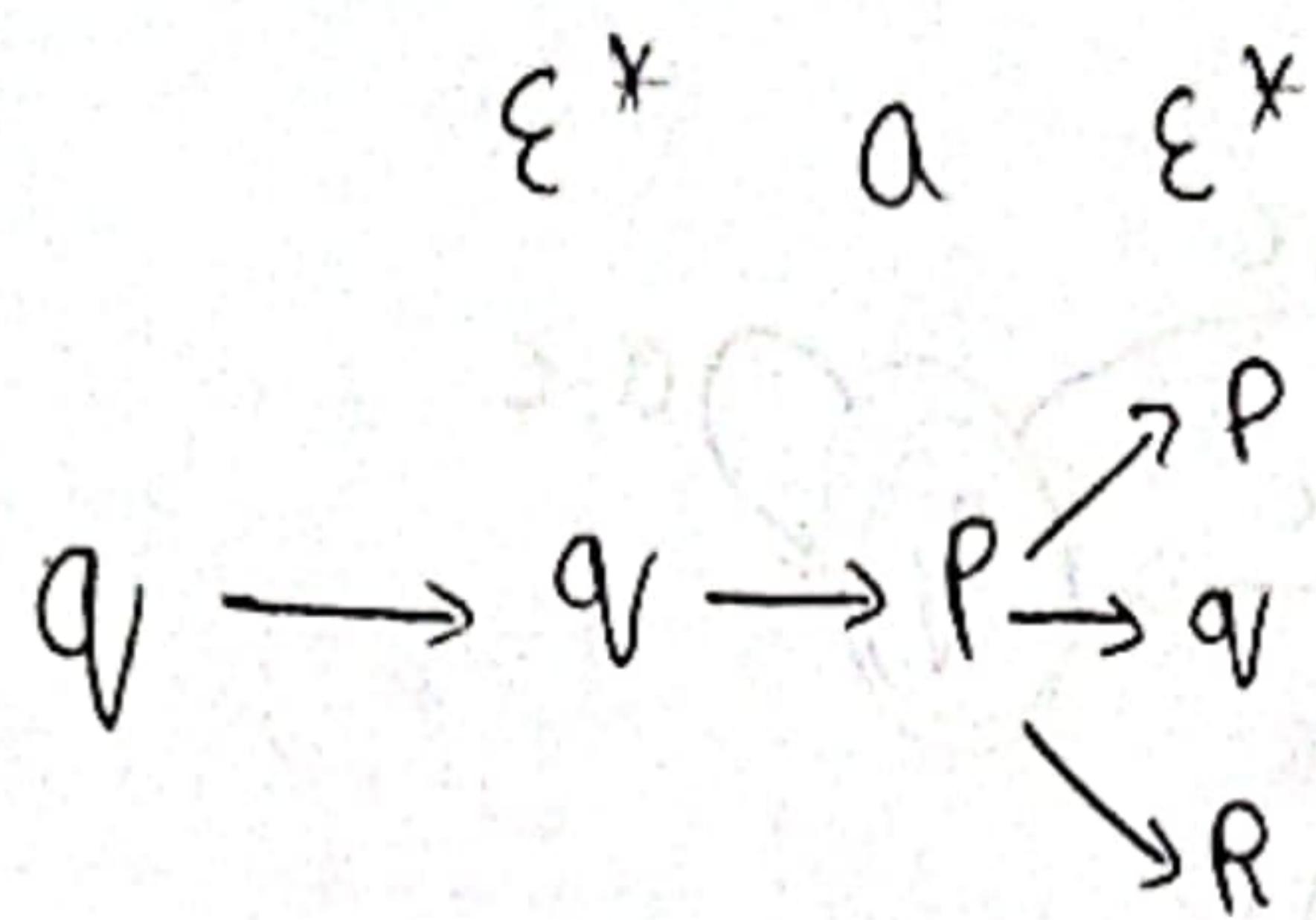


	0	1
$\rightarrow q_0$	$\{q_1, q_2, q_3\}$	$\{q_1, q_2, q_4\}$
$\times q_1$	$\{q_1, q_2, q_3\}$	$\{q_1, q_2, q_4\}$
$\times q_2$	$\{q_3\}$	$\{q_4\}$
q_3	$\{q_5, q_6, q_7\}$	\emptyset
q_4	\emptyset	$\{q_5, q_6, q_7\}$
$\times q_5$	$\{q_6, q_7\}$	$\{q_6, q_7\}$
$\times q_6$	$\{q_6, q_7\}$	$\{q_6, q_7\}$
$\times q_7$	\emptyset	\emptyset



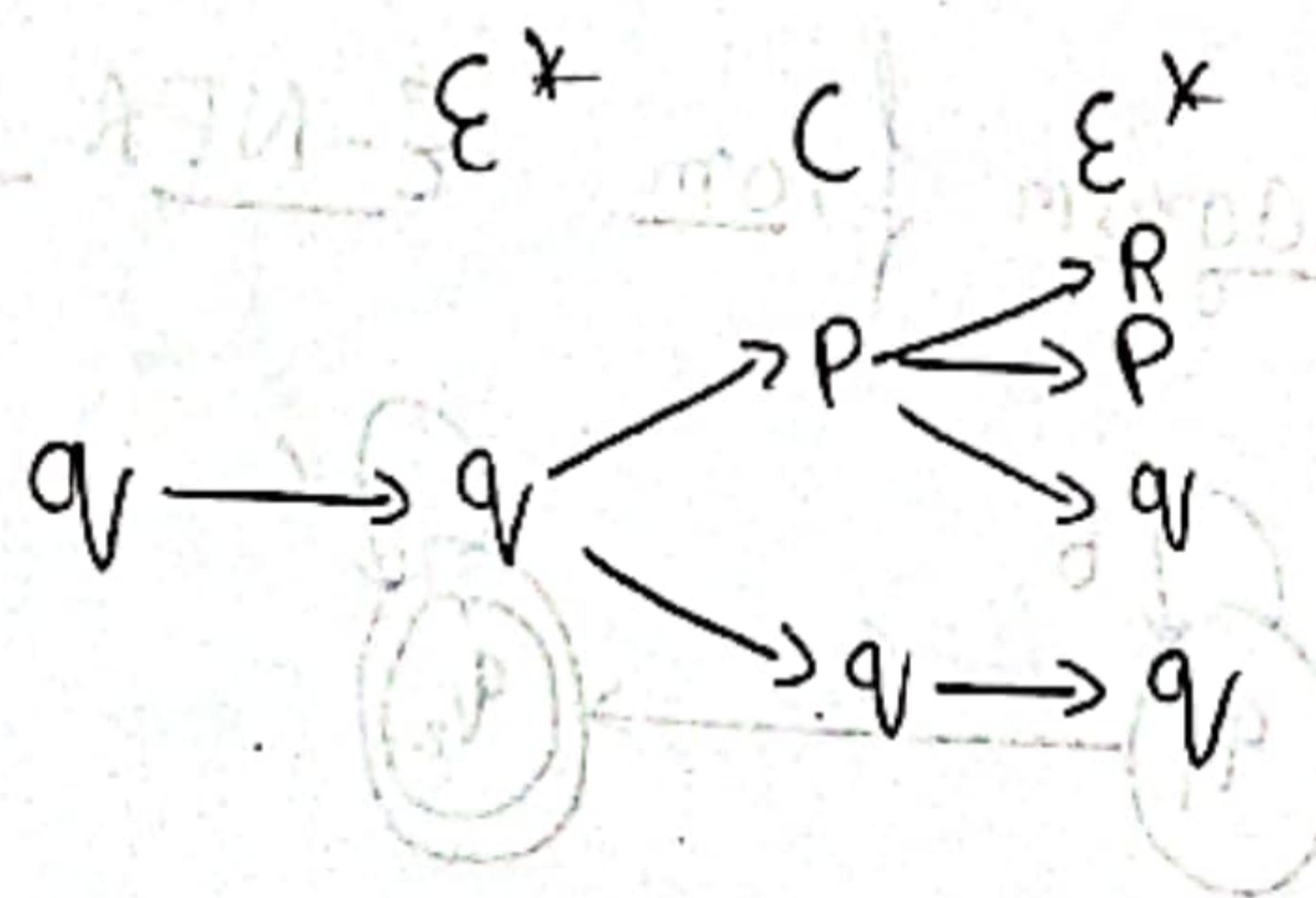
③ Convert the following into NFA:-





$\epsilon^* \quad b \quad \epsilon^*$

$q \xrightarrow{\epsilon^*} q \xrightarrow{R} R \xrightarrow{\epsilon^*} R$



$\epsilon^* \quad a \quad \epsilon^*$

$R \xrightarrow{\epsilon^*} R \xrightarrow{\phi}$

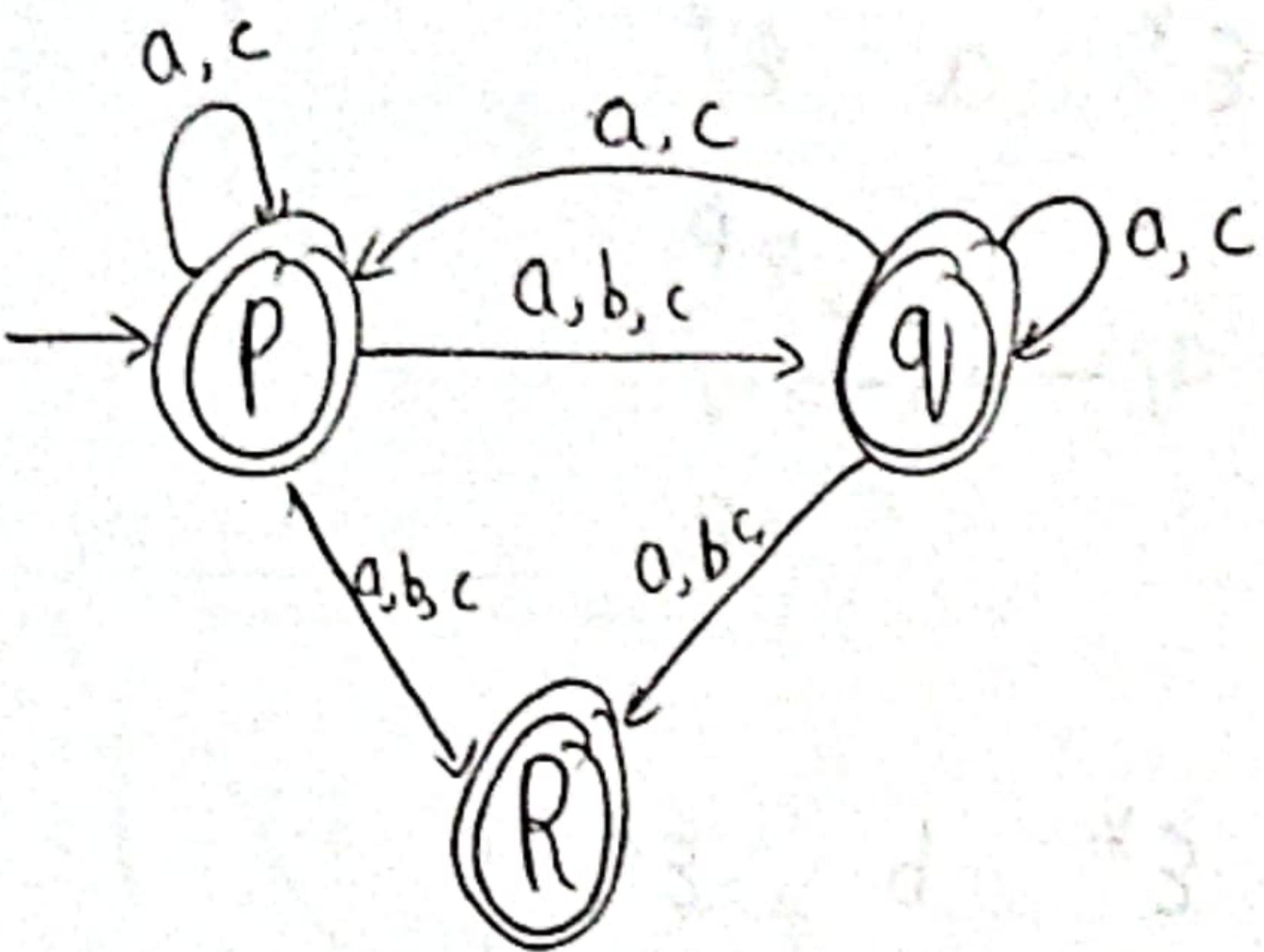
$\epsilon^* \quad b \quad \epsilon^*$

$R \xrightarrow{\epsilon^*} R \xrightarrow{\phi}$

$\epsilon^* \quad c \quad \epsilon^*$

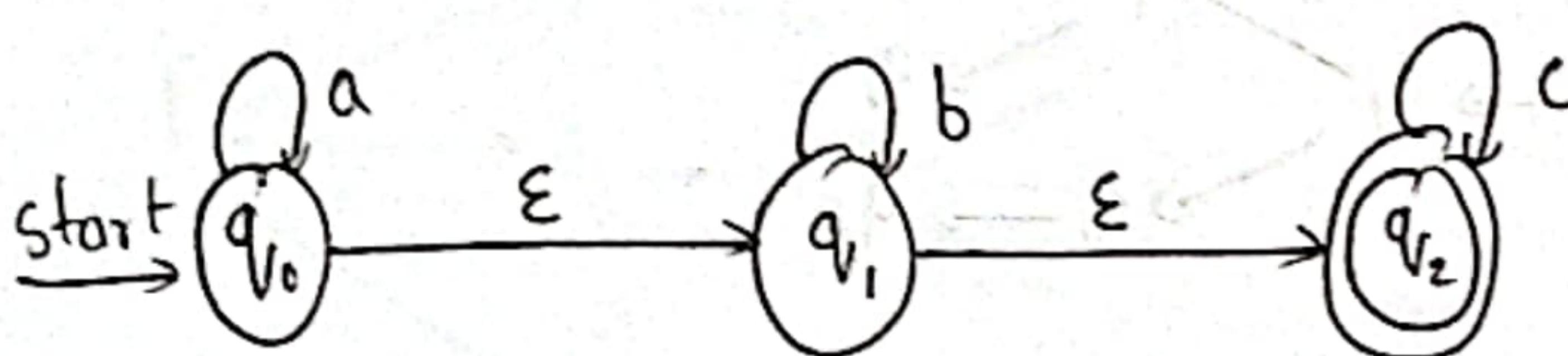
$R \xrightarrow{\epsilon^*} R \xrightarrow{\phi}$

	a	b	c
$\xrightarrow{*P}$	$\{p, q, R\}$	$\{q, R\}$	$\{p, q, R\}$
$\xrightarrow{*q}$	$\{p, q, R\}$	R	$\{p, q, R\}$
$\xrightarrow{*R}$	\emptyset	\emptyset	\emptyset



\Rightarrow Conversion from ϵ -NFA to DFA :-

① Convert the state diagram from ϵ -NFA to DFA :-



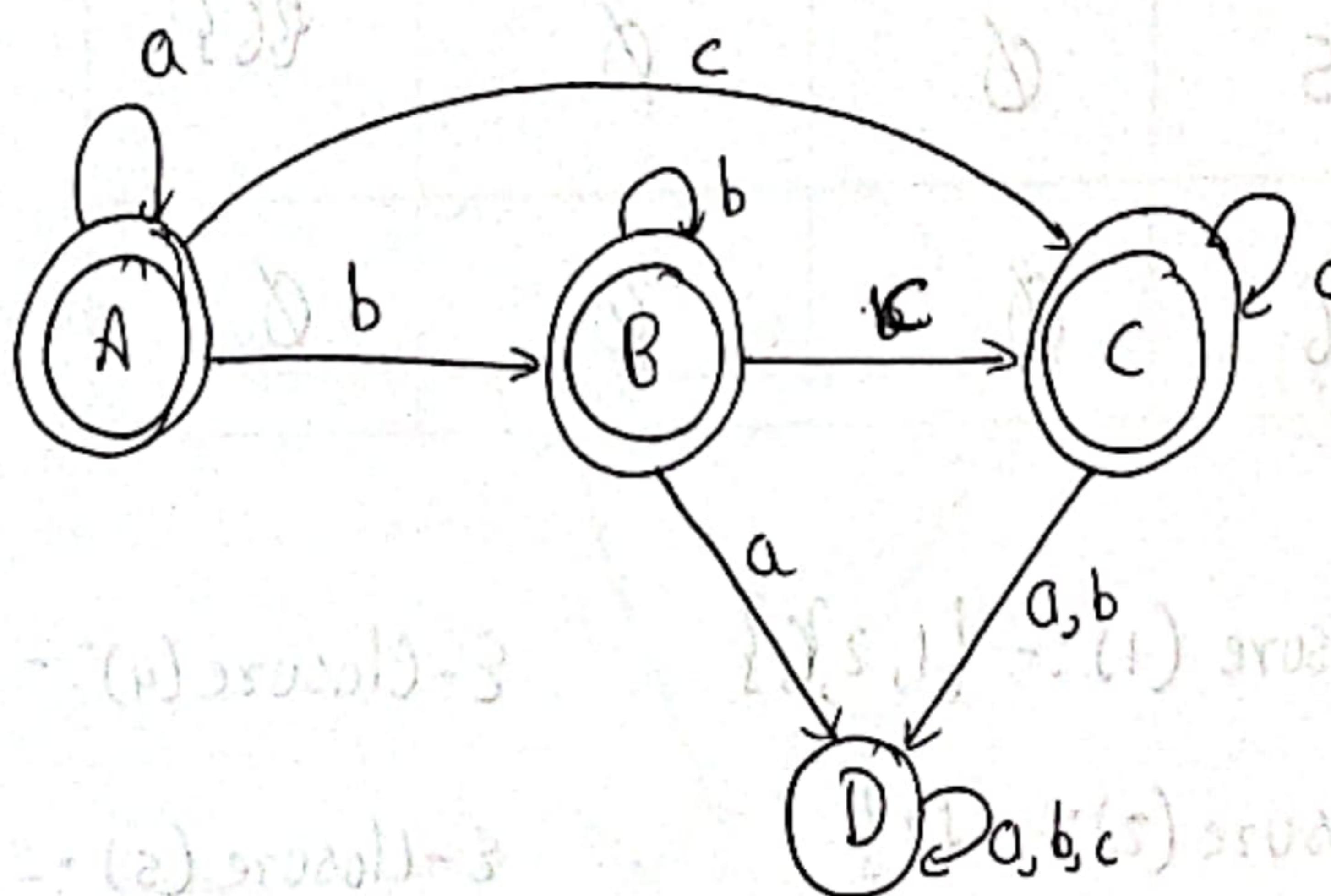
δ_ϵ	a	b	c	ϵ
$\rightarrow q_{v_0}$	{ q_{v_0} }	\emptyset	\emptyset	{ q_{v_1} }
q_{v_1}	\emptyset	{ q_{v_1} }	\emptyset	q_{v_2}
* q_{v_2}	\emptyset	\emptyset	{ q_{v_2} }	\emptyset

ϵ -closure (q_{v_0}) :- { $q_{v_0}, q_{v_1}, q_{v_2}$ }

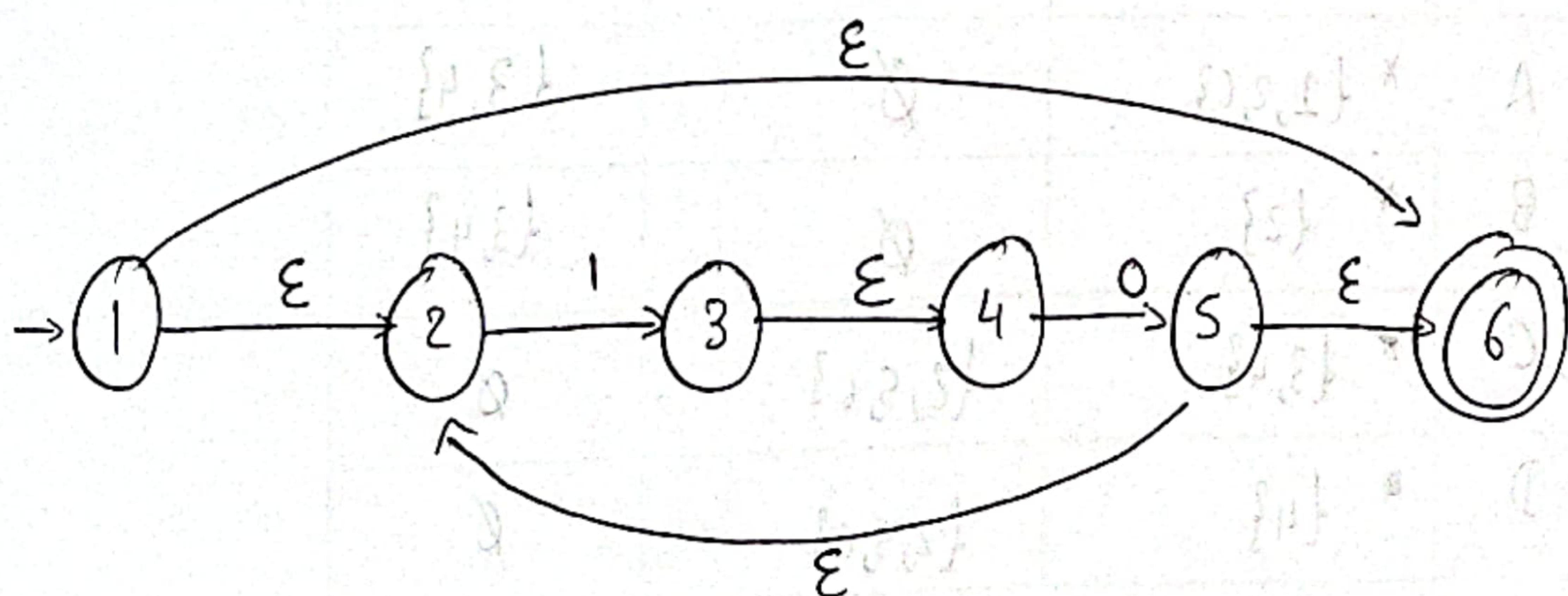
ϵ -closure (q_{v_1}) :- { q_{v_1}, q_{v_2} }

ϵ -closure (q_{v_2}) :- { q_{v_2} }

	δ_0	a	b	c
A \Leftarrow	$^*\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
B \Leftarrow	$^*\{q_1, q_2\}$	\emptyset	$\{q_1, q_2\}$	$\{q_2\}$
C \Leftarrow	$^*\{q_2\}$	\emptyset	\emptyset	$\{q_2\}$



② Convert ϵ -NFA to equivalent DFA :-



δ_ϵ	0	1	ϵ
1	\emptyset	\emptyset	{2,6}
2	\emptyset	3	\emptyset
3	\emptyset	\emptyset	{4}
4	5	\emptyset	\emptyset
5	\emptyset	\emptyset	{6}
6	\emptyset	\emptyset	\emptyset

ϵ -closure (1) :- {1, 2, 5}

ϵ -closure (4) :- {4}

ϵ -closure (2) :- {2}

ϵ -closure (5) :- {2, 5, 6}

ϵ -closure (3) :- {3, 4}

ϵ -closure (6) :- {6}

	δ_D	0	1
A	* {1, 2, 6}	\emptyset	{3, 4}
	* {2}	\emptyset	{3, 4}
B	* {3, 4}	{2, 5, 6}	\emptyset
	* {4}	{2, 5, 6}	\emptyset
C	* {2, 5, 6}	\emptyset	{3, 4}
D	* {6}	\emptyset	\emptyset

