CSE-233 : Section A

Summer 2020

ε−NFA

Reference:

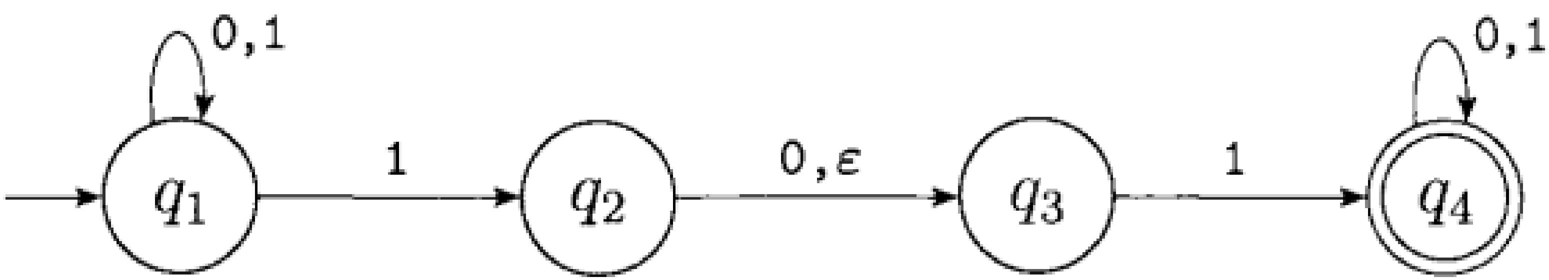
Book1 Chapter 2.5

Book2 Chapter 1.2

Md. Saidul Hoque Anik
anik@cse.uiu.ac.bd

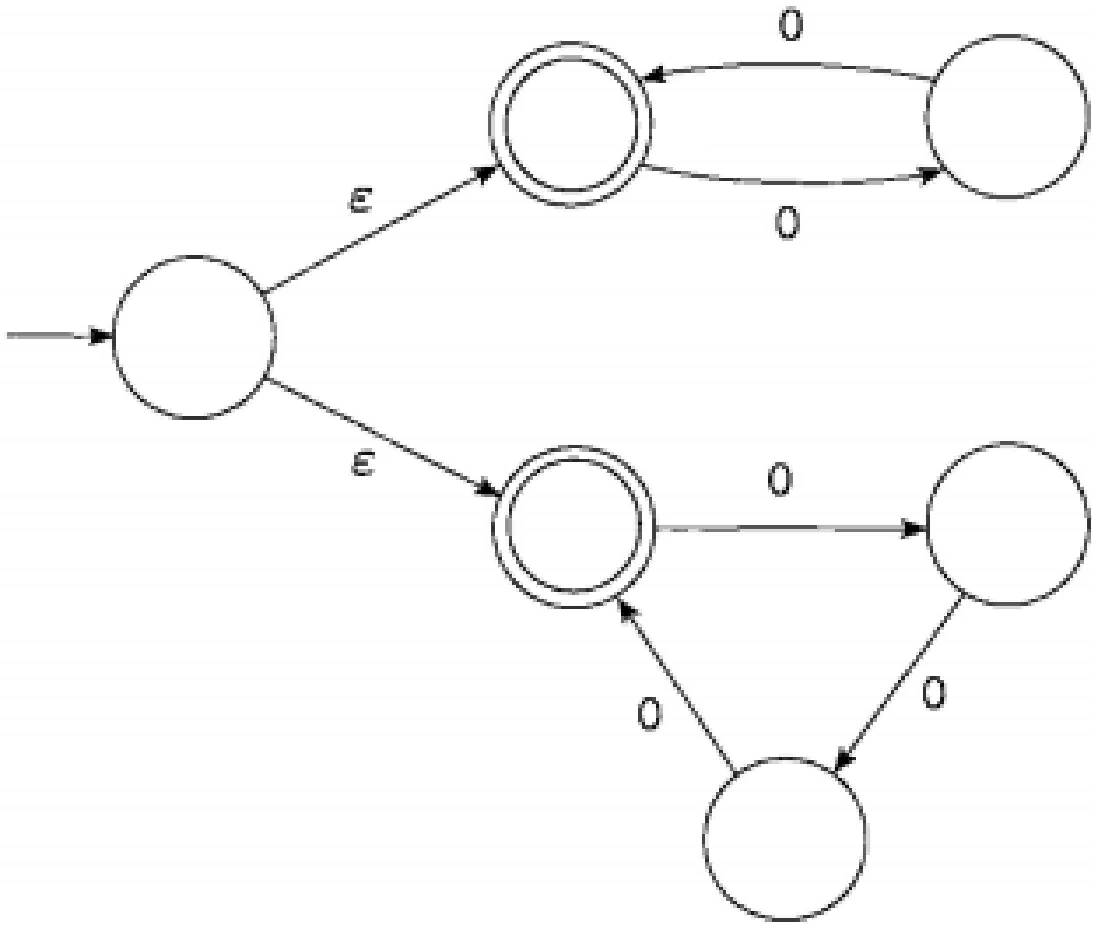
ε−NFA

- In NFA, if we come to a state with ε symbol on an exiting arrow, the machine immediately generates the next state as copy **before reading next input.**
- ε-NFA just adds a more flexible way to represent an NFA, it doesn't add new functionality. We can convert any ε-NFA to an equivalent NFA.



Usefulness

Following is a machine which recognizes all strings of the form 0^k where k is multiple of 2 or 3 over alphabet {0}



Usefulness

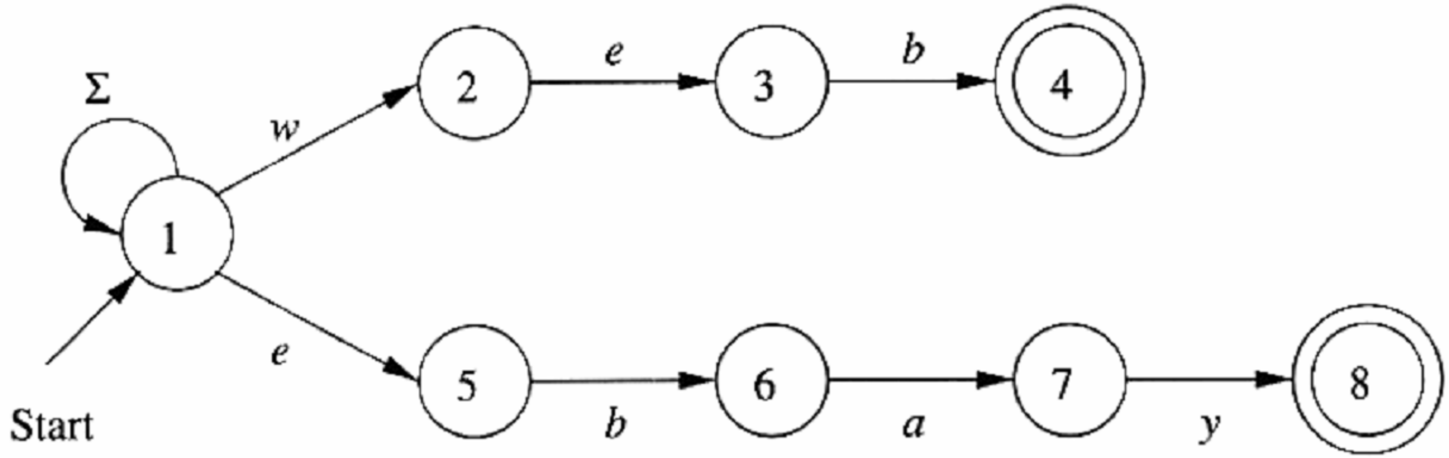


Figure 2.16: An NFA that searches for the words **web** and **ebay**

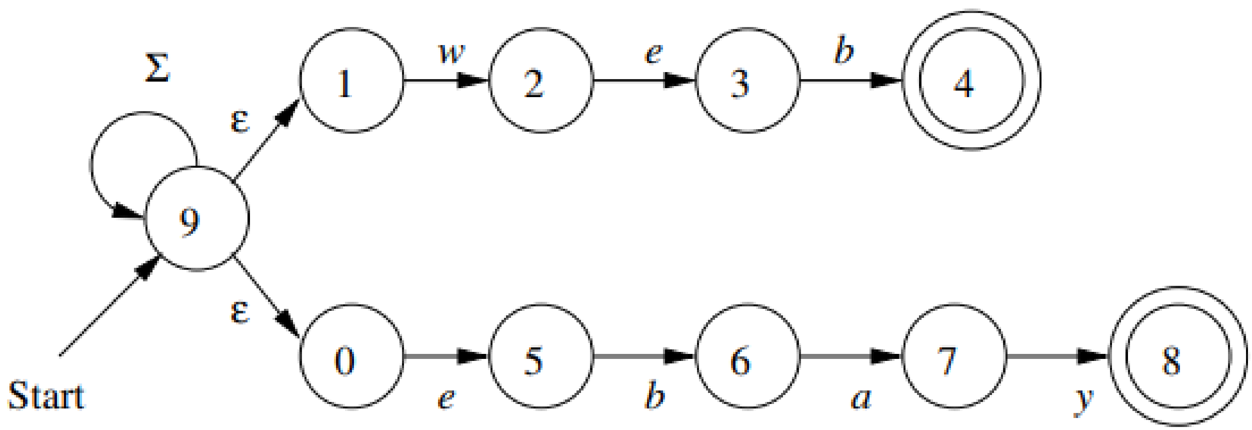
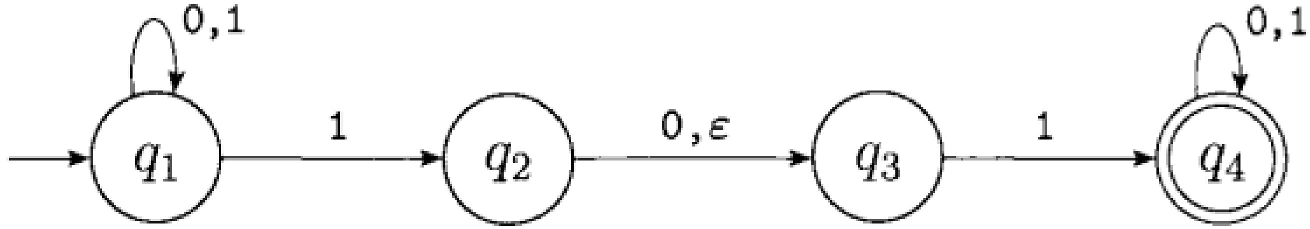
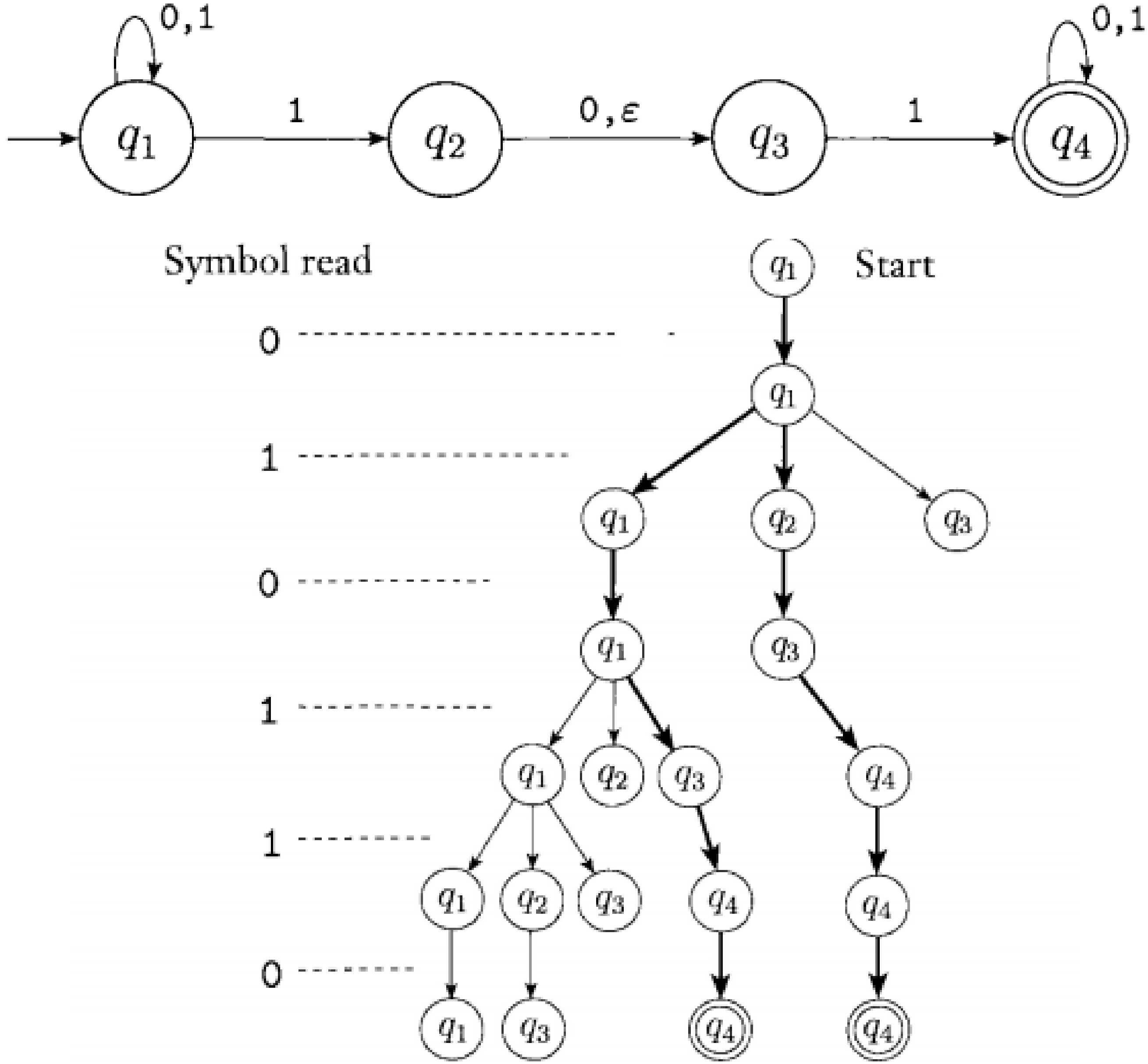


Figure 2.19: Using ϵ-transitions to help recognize keywords

State transition tree for input: 010110

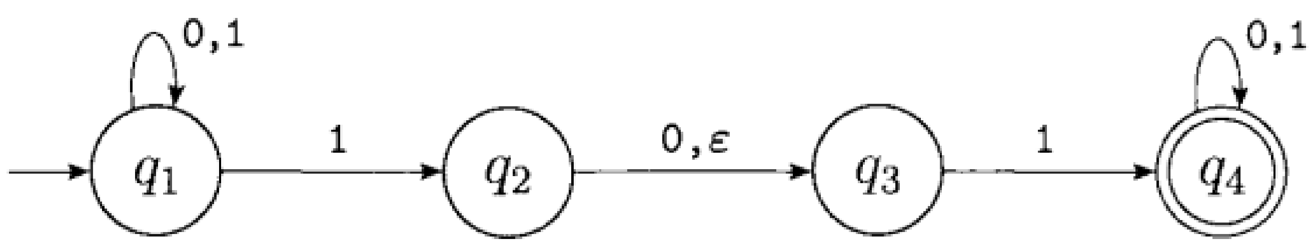


State transition tree for input: 010110



ε-closures

- Notice that whenever we arrive at state q_2, we see q_3 gets generated with it because it has ε to q_3. So we say ECLOSE(q_2) = {q_2, q_3}
- ECLOSE(q) = Every state that can be reached from q along any path whose arcs (arrows) are all labeled ε and q itself.
- Also denoted as ε*



Example

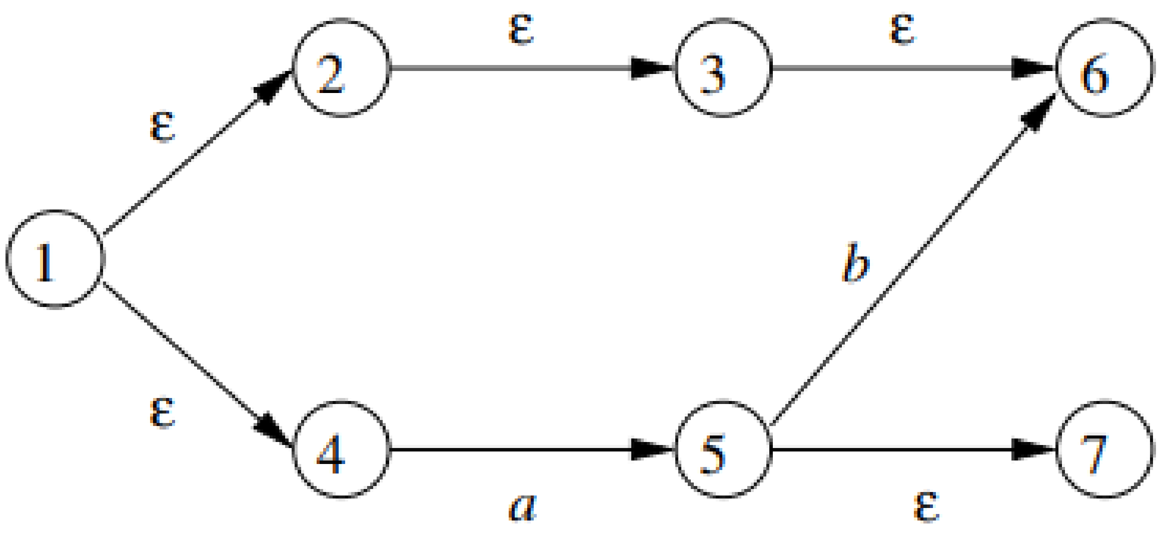


Figure 2.21: Some states and transitions

ECLOSE(1) = ?

Example

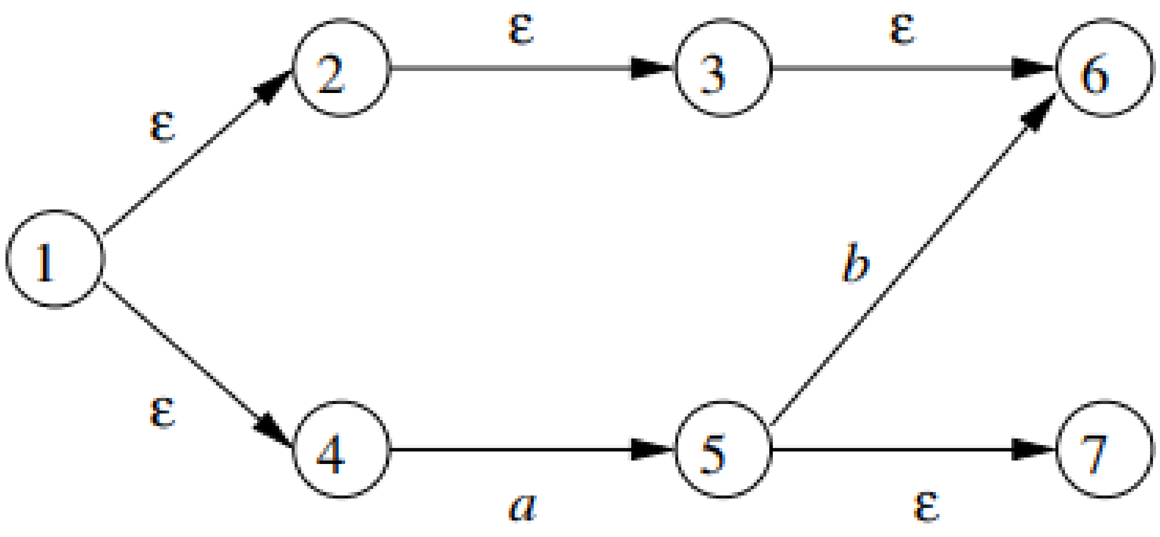


Figure 2.21: Some states and transitions

ECLOSE(1) = {1, 2, 3, 6, 4}

Example

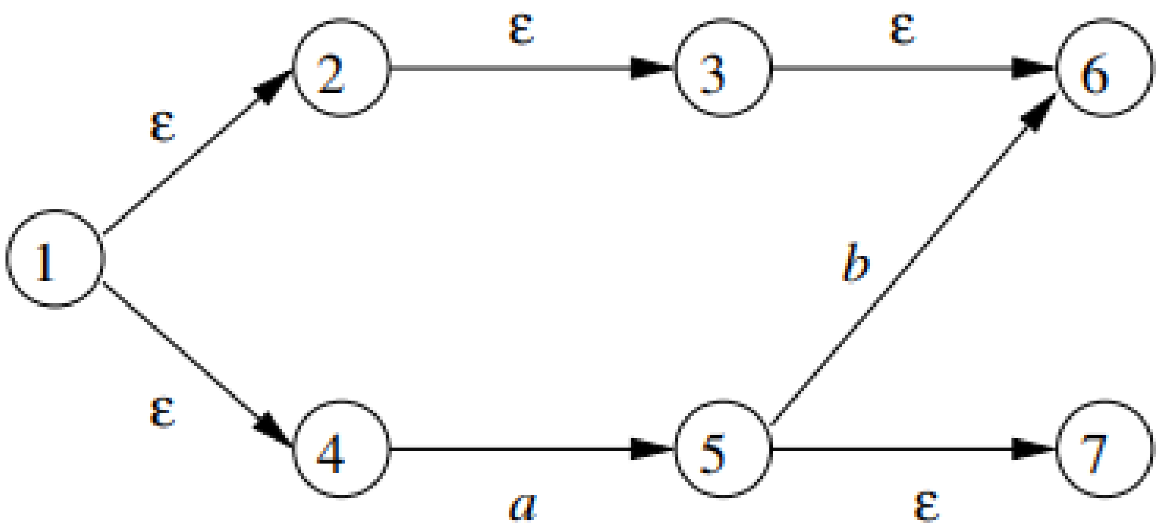


Figure 2.21: Some states and transitions

ECLOSE(1) = {1, 2, 3, 4, 6}

ECLOSE(2) = ?

ECLOSE(3) = ?

ECLOSE(4) = ?

ECLOSE(5) = ?

ECLOSE(6) = ?

ECLOSE(7) = ?

Significance

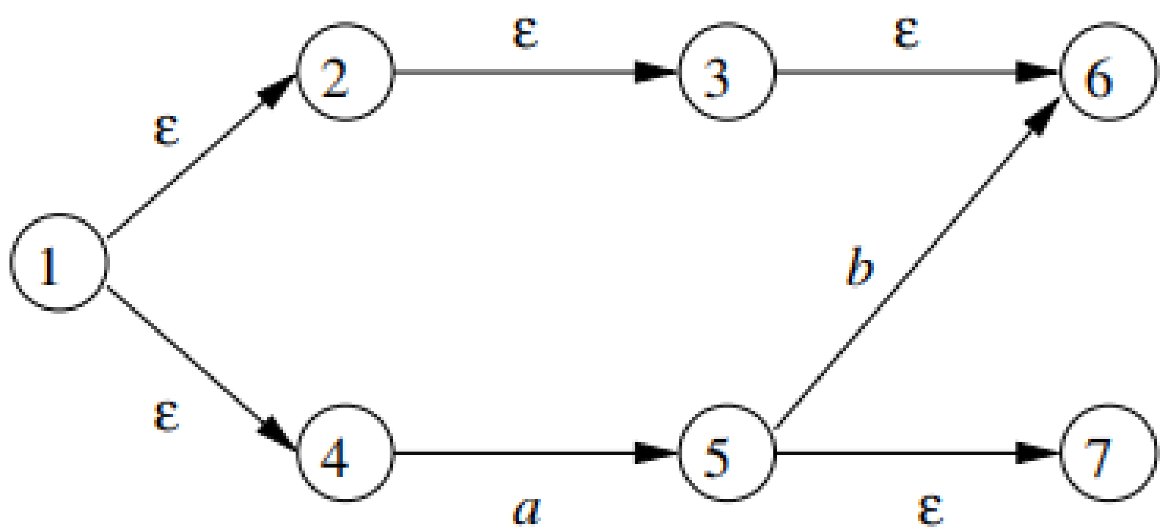


Figure 2.21: Some states and transitions

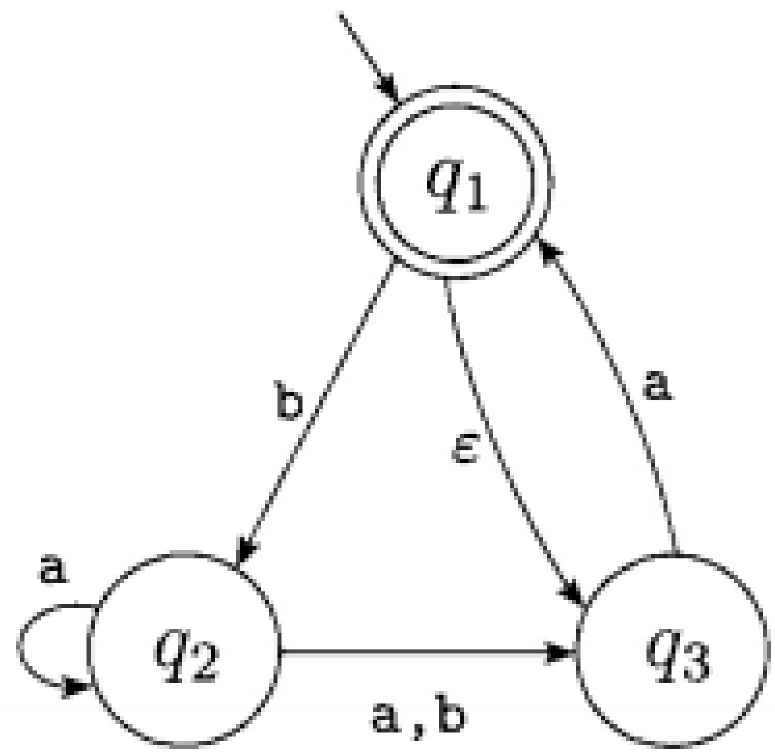
ECLOSE(1) = {1, 2, 3, 4, 6}

ε-closures tells us the actual states we shall have when we visit a state. For example, if we visit 1, we are having 2, 3, 4 and 6 also present that the same scene.

Task

Draw the state transition tree of this NFA for the following input.

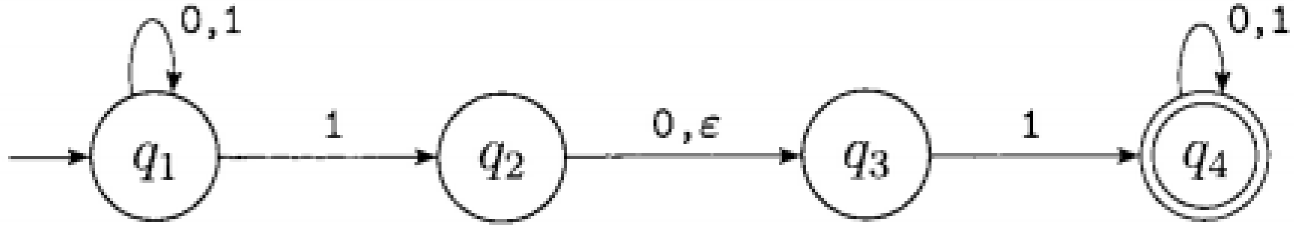
- baba
- baa
- babba
- bb



Does it accept?

- Empty String?
- a?
- b

Formal Notation of ε-NFA



The formal description of N_1 is $(Q, \Sigma, \delta, q_1, F)$, where

1. $Q = \{q_1, q_2, q_3, q_4\}$,

2. $\Sigma = \{0,1\}$,

3. δ is given as

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

,

4. q_1 is the start state, and

5. $F = \{q_4\}$.

Formal Notation of ε-NFA

An ϵ-NFA $A = (Q, \Sigma, \delta, q_0, F)$ where all components have their same interpretation as for NFA, except that δ is now a function that takes arguments:

1. A state in Q and
2. A member of $\Sigma \cup \{\epsilon\}$
   We require that ϵ cannot be a member of Σ

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

,

A More Complex Example

Example 2.16: In Fig. 2.18 is an ϵ-NFA that accepts decimal numbers consisting of:

1. An optional $+$ or $-$ sign,
2. A string of digits,
3. A decimal point, and
4. Another string of digits. Either this string of digits, or the string (2) can be empty, but at least one of the two strings of digits must be nonempty.

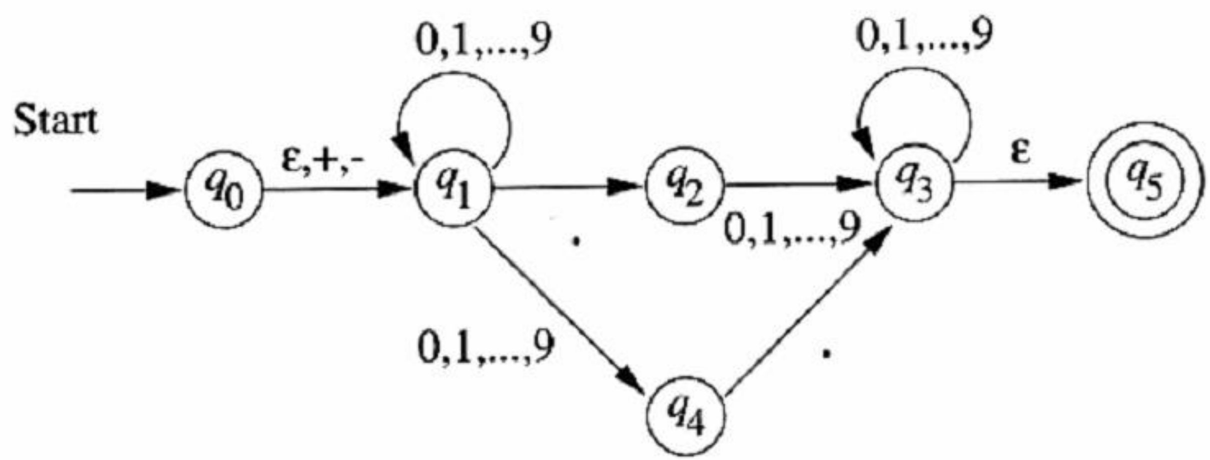


Figure 2.18: An ϵ-NFA accepting decimal numbers

Transition Table for the example

Example 2.18 : The ϵ-NFA of Fig. 2.18 is represented formally as

$$E = (\{q_0, q_1, \ldots, q_5\}, \{., +, -, 0, 1, \ldots, 9\}, \delta, q_0, \{q_5\})$$

where δ is defined by the transition table in Fig. 2.20. □

	ϵ	$+, -$	$.$	$0, 1, \ldots, 9$
q_0	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_3	$\{q_5\}$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
q_5	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Figure 2.20: Transition table for Fig. 2.18

Extended Transition Function

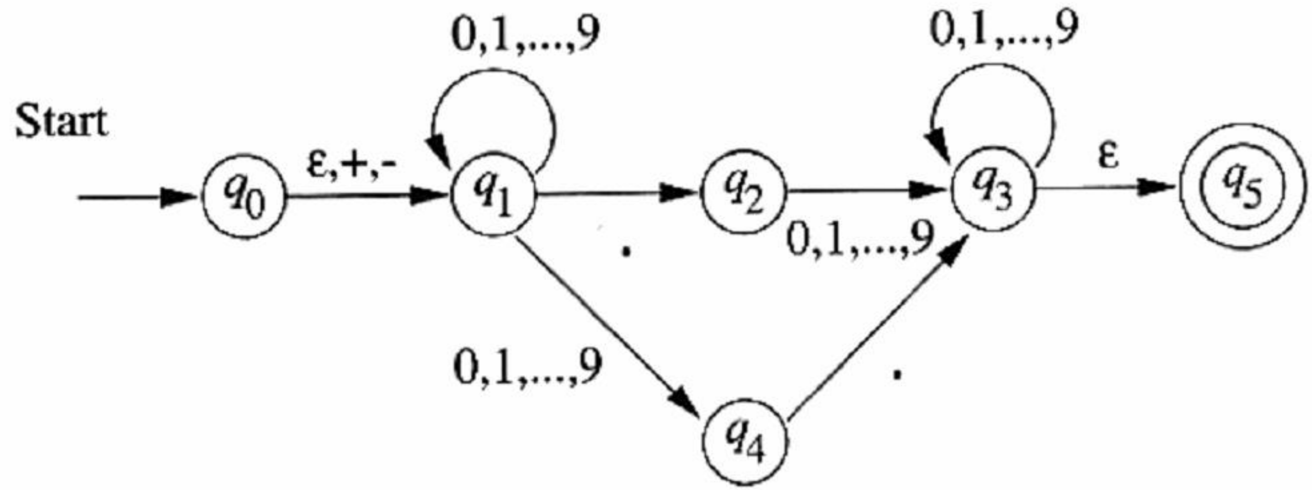


Figure 2.18: An ϵ-NFA accepting decimal numbers

Let us compute $\hat{\delta}(q_0, 5.6)$ for the ϵ-NFA of Fig. 2.18.

Extended Transition Function

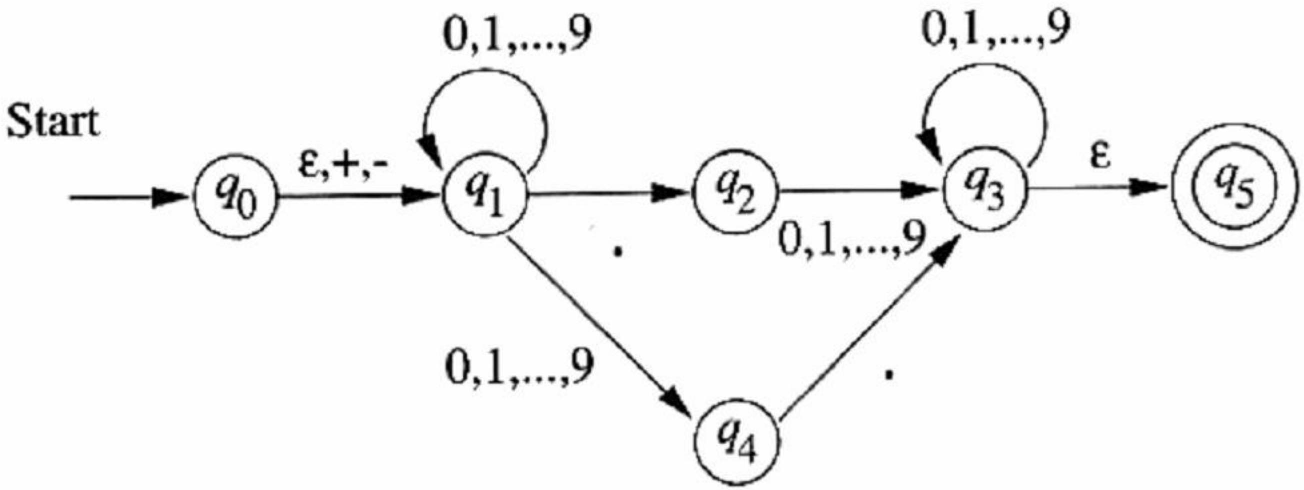


Figure 2.18: An ϵ-NFA accepting decimal numbers

Example 2.20: Let us compute $\hat{\delta}(q_0, \texttt{5.6})$ for the ϵ-NFA of Fig. 2.18. A summary of the steps needed are as follows:

- $\hat{\delta}(q_0, \epsilon) = \text{ECLOSE}(q_0) = \{q_0, q_1\}$.

Extended Transition Function

- Compute $\hat{\delta}(q_0, 5)$ as follows:

  1. First compute the transitions on input 5 from the states q_0 and q_1 that we obtained in the calculation of $\hat{\delta}(q_0, \epsilon)$, above. That is, we compute $\delta(q_0, 5) \cup \delta(q_1, 5) = \{q_1, q_4\}$.
  2. Next, ϵ-close the members of the set computed in step (1). We get $\text{ECLOSE}(q_1) \cup \text{ECLOSE}(q_4) = \{q_1\} \cup \{q_4\} = \{q_1, q_4\}$. That set is $\hat{\delta}(q_0, 5)$. This two-step pattern repeats for the next two symbols.

- Compute $\hat{\delta}(q_0, 5.)$ as follows:

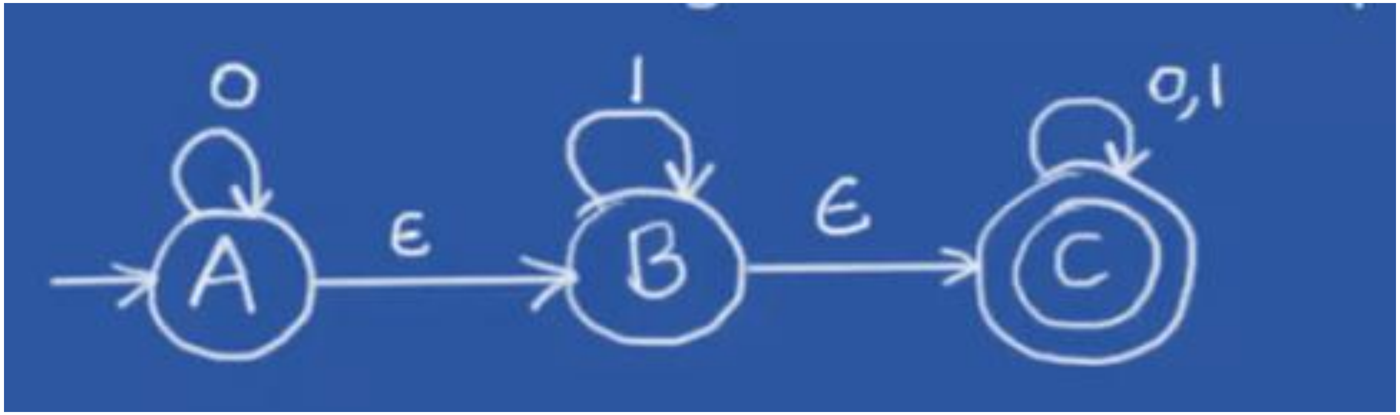
  1. First compute $\delta(q_1, .) \cup \delta(q_4, .) = \{q_2\} \cup \{q_3\} = \{q_2, q_3\}$.
  2. Then compute

$$\hat{\delta}(q_0, 5.) = \text{ECLOSE}(q_2) \cup \text{ECLOSE}(q_3) = \{q_2\} \cup \{q_3, q_5\} = \{q_2, q_3, q_5\}$$

- Compute $\hat{\delta}(q_0, 5.6)$ as follows:

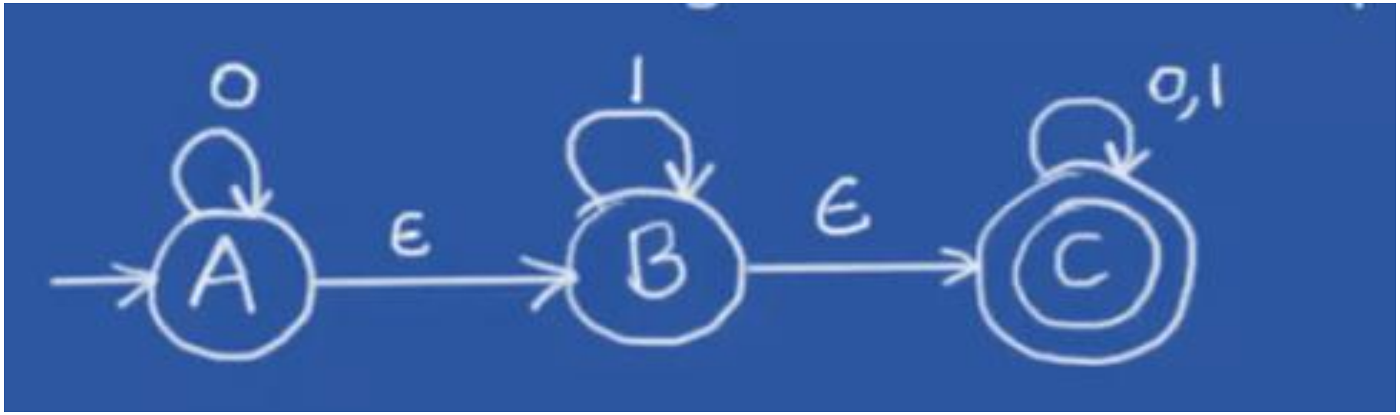
  1. First compute $\delta(q_2, 6) \cup \delta(q_3, 6) \cup \delta(q_5, 6) = \{q_3\} \cup \{q_3\} \cup \emptyset = \{q_3\}$.
  2. Then compute $\hat{\delta}(q_0, 5.6) = \text{ECLOSE}(q_3) = \{q_3, q_5\}$.

Conversion from ε–NFA to NFA



Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε−NFA to NFA

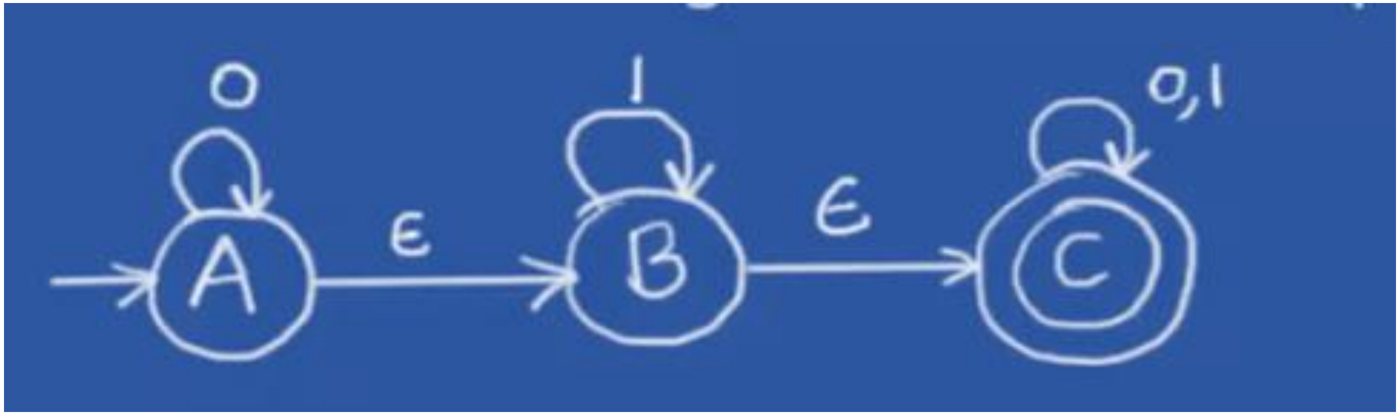


Step 1: Calculate ECLOSE() for all states

State	ECLOSE()
A	
B	
C	

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε−NFA to NFA

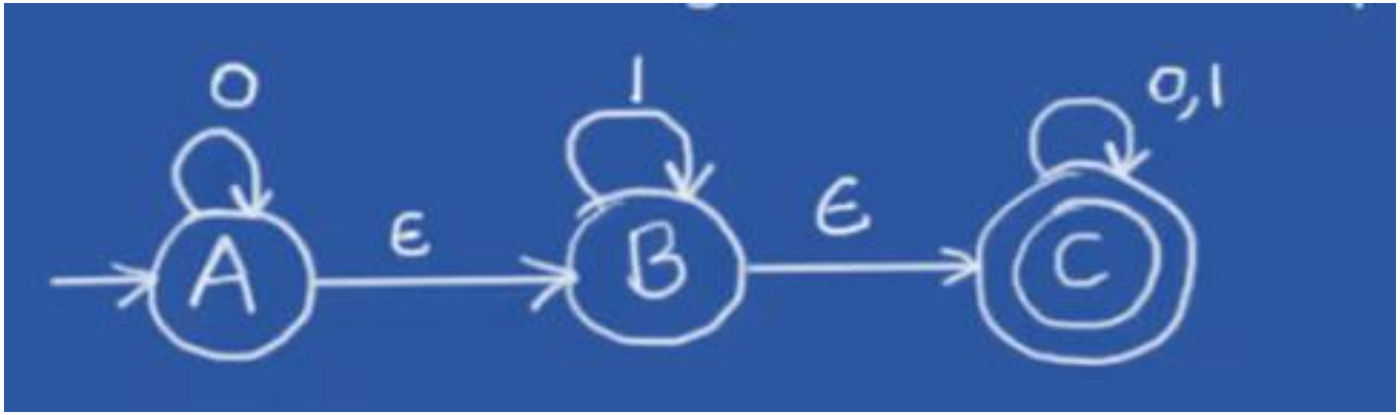


Step 1: Calculate ECLOSE() for all states

State	ECLOSE()
A	{A, B, C}
B	
C	

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε−NFA to NFA

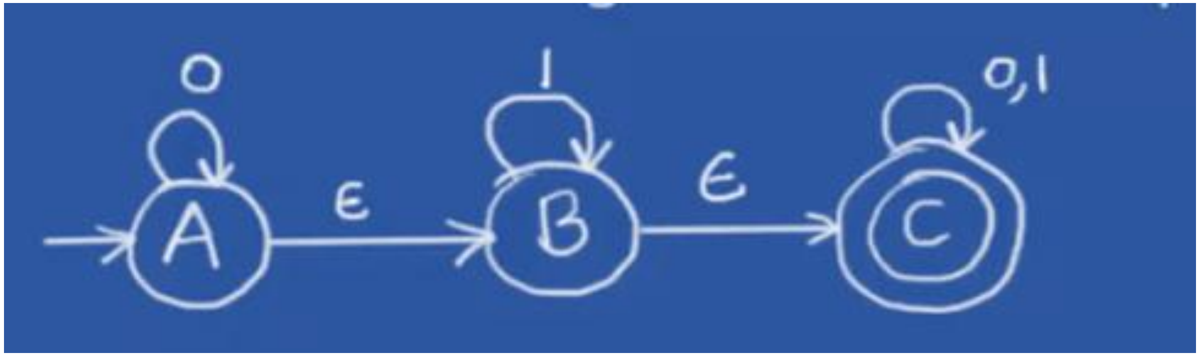


Step 1: Calculate ECLOSE() for all states

State	ECLOSE()
A	{A, B, C}
B	{B, C}
C	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε–NFA to NFA



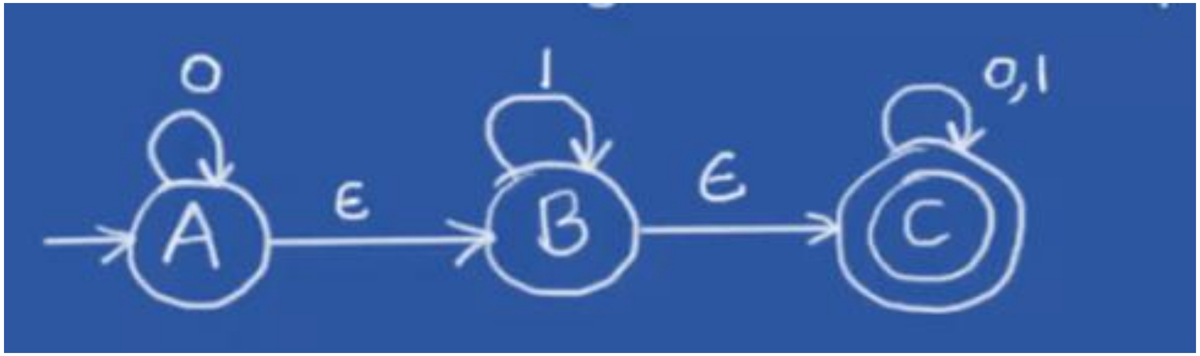
Step 2: For each state q, apply input on ECLOSE(q) and the next states will be the ECLOSE of each of the resulting states.

State	ECLOSE()	0	ECLOSE()	1	ECLOSE()
A					
B					
C					

State	ECLOSE()
A	{A, B, C}
B	{B, C}
C	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε-NFA to NFA



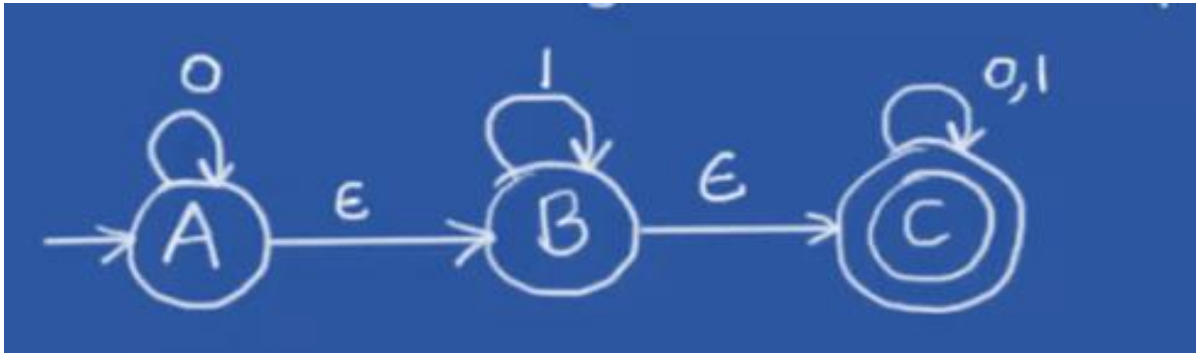
Step 2: For each state q, apply input on ECLOSE(q) and the next states will be the ECLOSE of each of the resulting states.

State	ECLOSE()	0	ECLOSE()	1	ECLOSE()
A	{A, B, C}				
B					
C					

State	ECLOSE()
A	{A, B, C}
B	{B, C}
C	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε−NFA to NFA



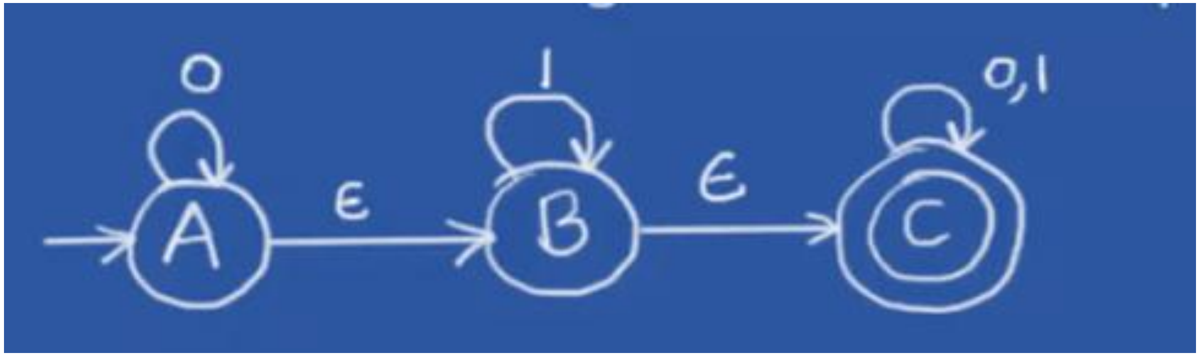
Step 2: For each state q, apply input on ECLOSE(q) and the next states will be the ECLOSE of each of the resulting states.

State	ECLOSE()	0	ECLOSE()	1	ECLOSE()
A	{A, B, C}	{A,C}	{A,B,C}	{B,C}	{B,C}
B	{B, C}				
C					

State	ECLOSE()
A	{A, B, C}
B	{B, C}
C	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε−NFA to NFA



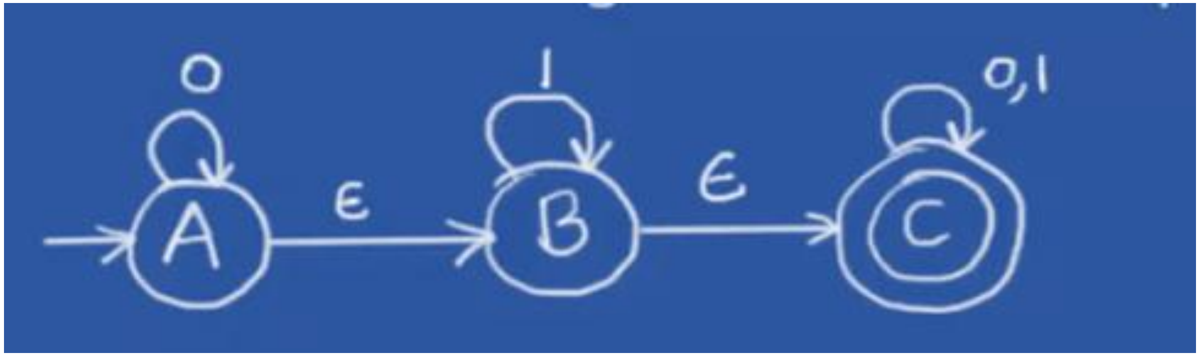
Step 2: For each state q, apply input on ECLOSE(q) and the next states will be the ECLOSE of each of the resulting states.

State	ECLOSE()	0	ECLOSE()	1	ECLOSE()
A	{A, B, C}	{A,C}	{A,B,C}	{B,C}	{B,C}
B	{B, C}	{C}	{C}	{B,C}	{B,C}
C	{C}				

State	ECLOSE()
A	{A, B, C}
B	{B, C}
C	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε–NFA to NFA



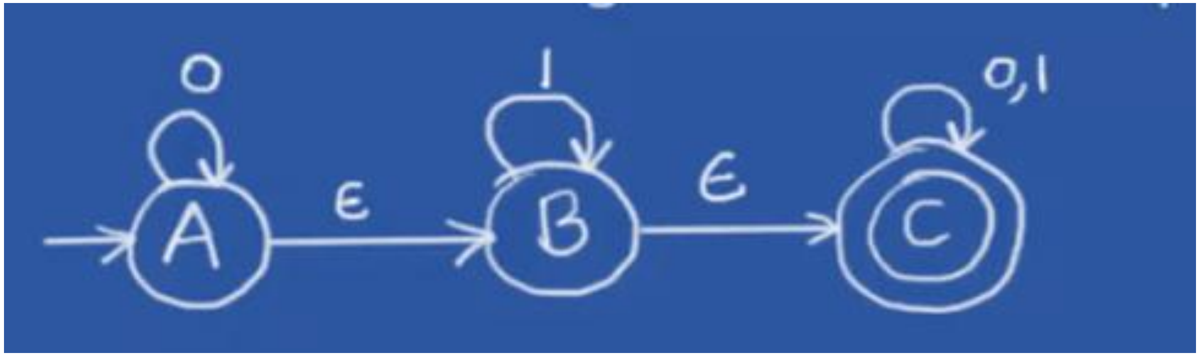
Step 2: For each state q, apply input on ECLOSE(q) and the next states will be the ECLOSE of each of the resulting states.

State	ECLOSE()	0	ECLOSE()	1	ECLOSE()
A	{A, B, C}	{A,C}	{A,B,C}	{B,C}	{B,C}
B	{B, C}	{C}	{C}	{B,C}	{B,C}
C	{C}	{C}	{C}	{C}	{C}

State	ECLOSE()
A	{A, B, C}
B	{B, C}
C	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε–NFA to NFA

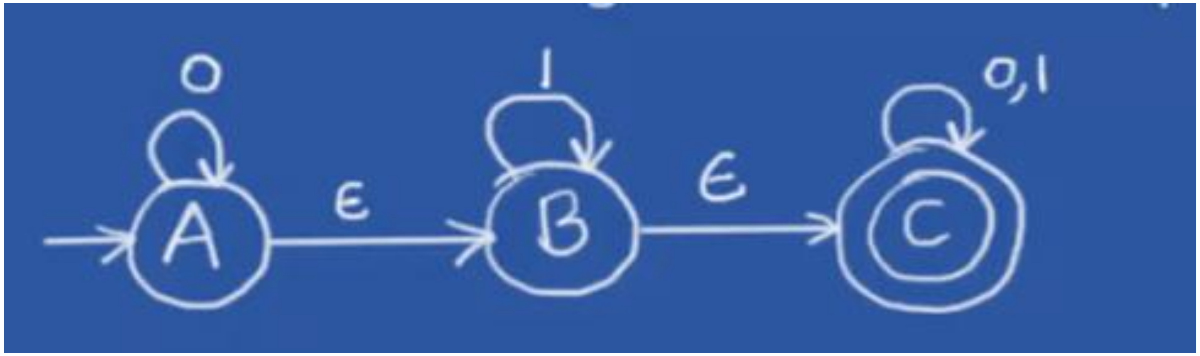


The ECLOSE of each of the resulting states is the final output of NFA.

State	ECLOSE()	0	1
A	{A, B, C}	{A,B,C}	{B,C}
B	{B, C}	{C}	{B,C}
C	{C}	{C}	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Conversion from ε–NFA to NFA

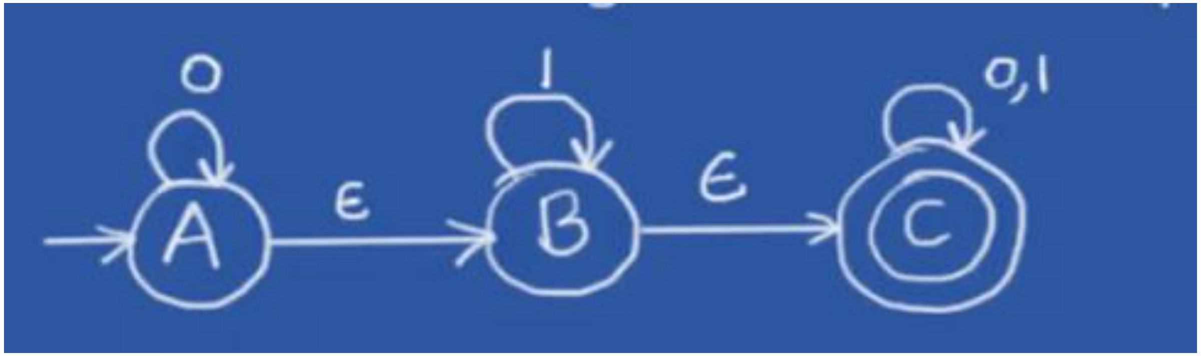


If the ECLOSE of a state contains final state, then it's the final state itself. The starting state will be the same.

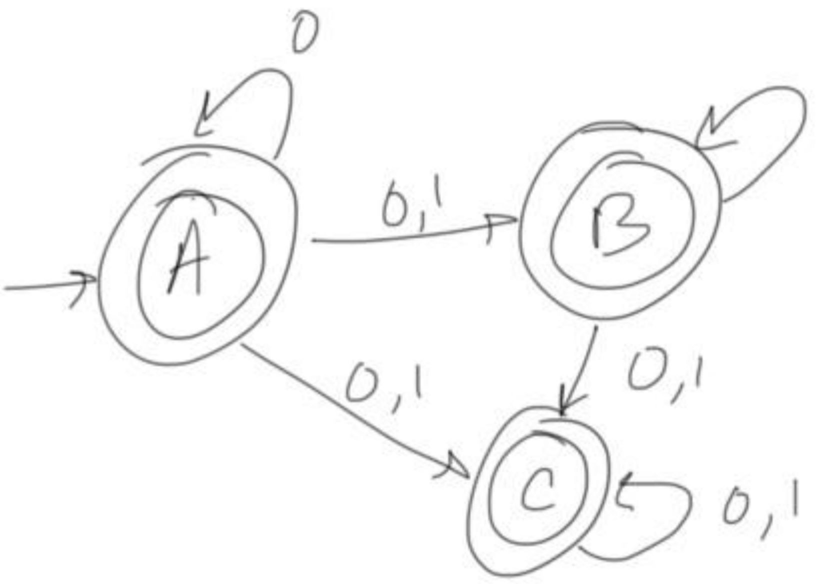
State	ECLOSE()	0	1
→*A	{A, B, C}	{A,B,C}	{B,C}
*B	{B, C}	{C}	{B,C}
*C	{C}	{C}	{C}

Reference: https://www.youtube.com/watch?v=WSGcmaHNBFM

Final NFA



State	0	1
→*A	{A,B,C}	{B,C}
*B	{C}	{B,C}
*C	{C}	{C}



Conversion from ε–NFA to DFA?

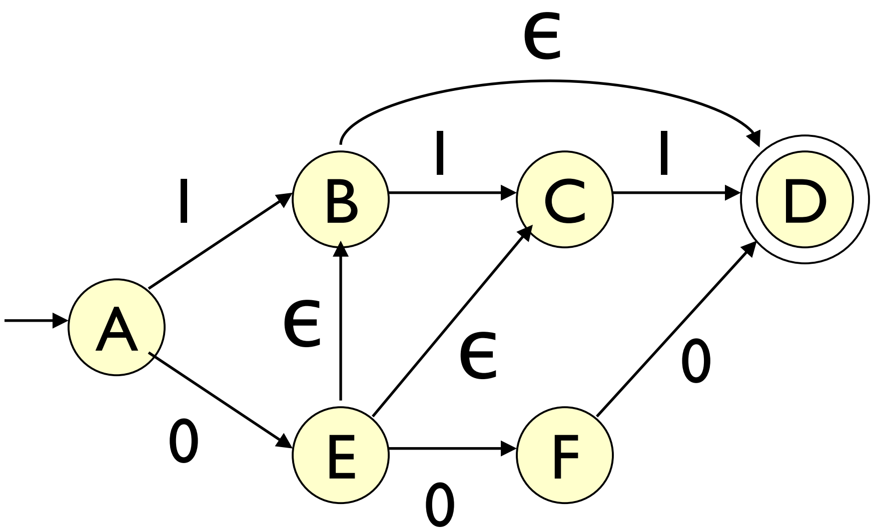
We can use Subset Construction method while building the NFA to DFA

Or, We can convert in step-by-step manner **(preferred)**:

ε-NFA → NFA → DFA

Task

Convert the following ϵ-NFA to NFA.



		0	1	ϵ
→	A	{E}	{B}	∅
	B	∅	{C}	{D}
	C	∅	{D}	∅
*	D	∅	∅	∅
	E	{F}	∅	{B, C}
	F	{D}	∅	∅

Solution

ECLOSE(B) = {B,D}
ECLOSE(E) = {B,C,D,E}

Task

Convert the following ϵ-NFA to DFA.

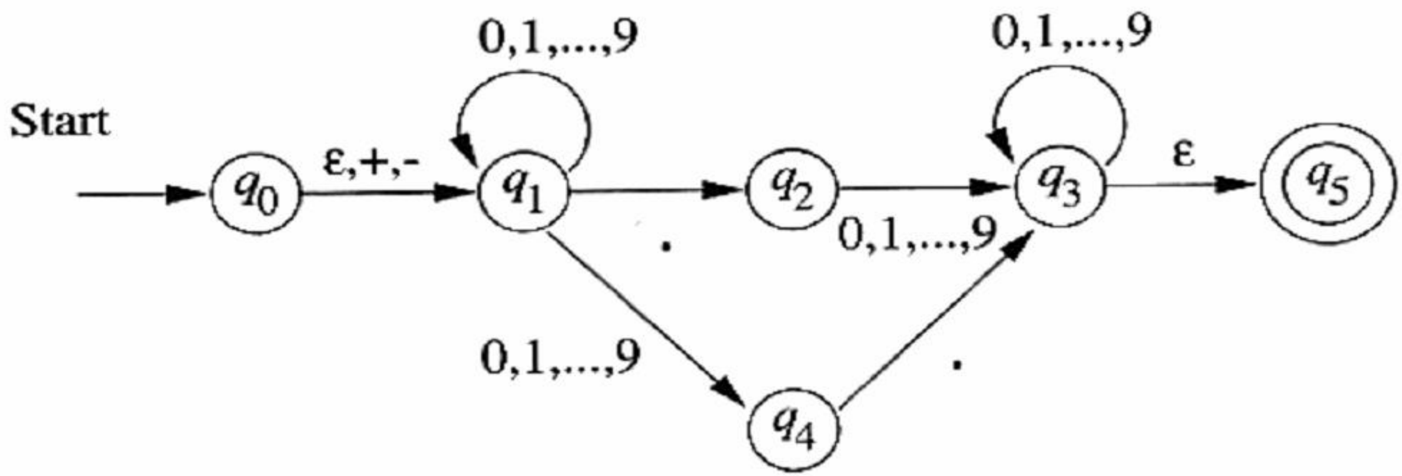


Figure 2.18: An ϵ-NFA accepting decimal numbers

Task

Convert the following ϵ-NFA to DFA.

Example 2.18: The ϵ-NFA of Fig. 2.18 is represented formally as

$$E = (\{q_0, q_1, \ldots, q_5\}, \{., +, -, 0, 1, \ldots, 9\}, \delta, q_0, \{q_5\})$$

where δ is defined by the transition table in Fig. 2.20. □

	ϵ	$+, -$	$.$	$0, 1, \ldots, 9$
q_0	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_3	$\{q_5\}$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
q_5	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Figure 2.20: Transition table for Fig. 2.18

Solution

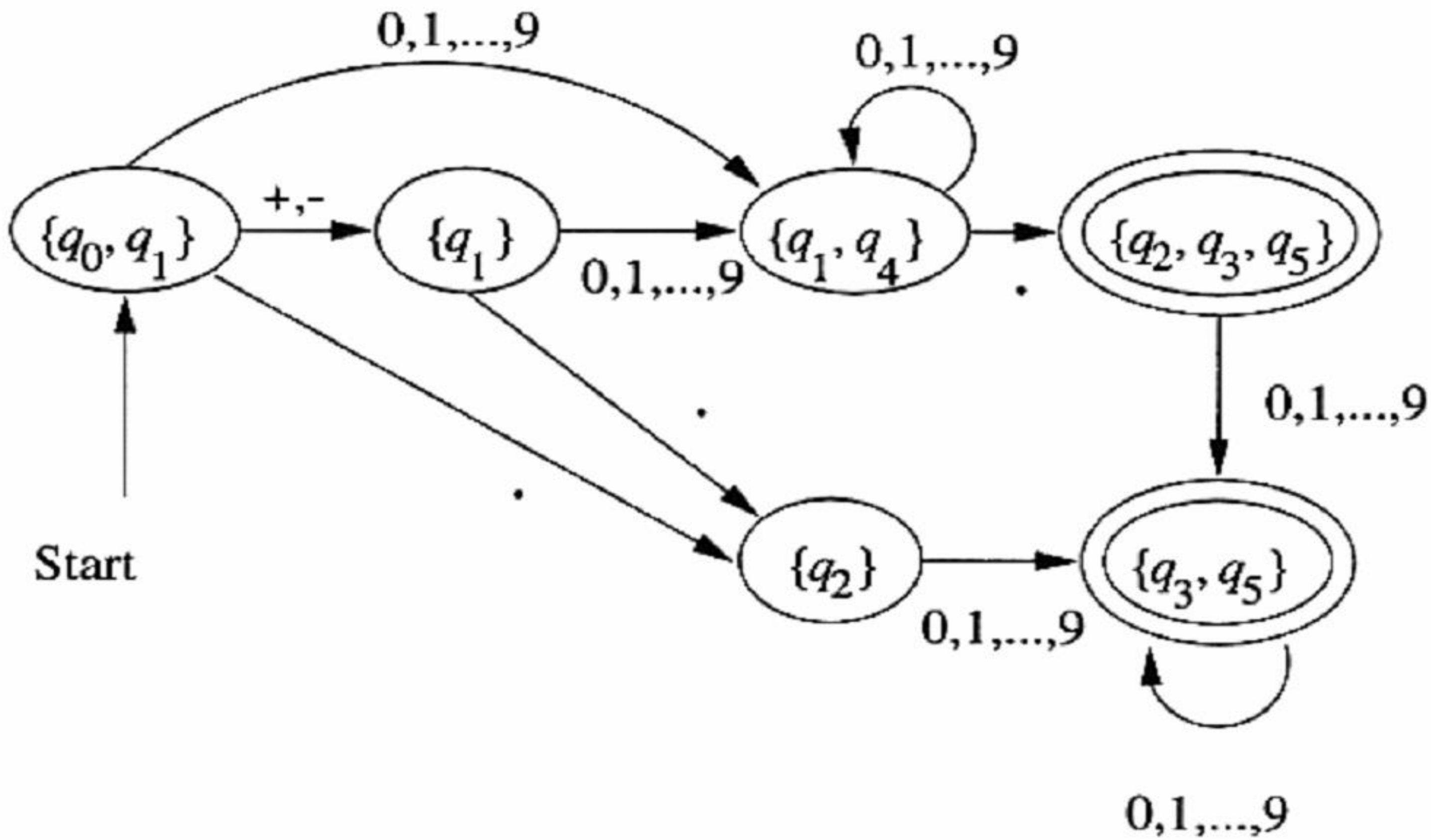


Figure 2.22: The DFA D that eliminates ϵ-transitions from Fig. 2.18

Practice

Exercise 2.5.1: Consider the following ϵ-NFA.

	ϵ	a	b	c
$\rightarrow p$	$\emptyset$	$\{p\}$	$\{q\}$	$\{r\}$
q	$\{p\}$	$\{q\}$	$\{r\}$	$\emptyset$
$*r$	$\{q\}$	$\{r\}$	$\emptyset$	$\{p\}$

a) Compute the ϵ-closure of each state.

b) Give all the strings of length three or less accepted by the automaton.

c) Convert the automaton to a DFA.