Automata

Lecture 4

Dr.Ebtsam Abdelhakam Eldesoky

The Story So Far

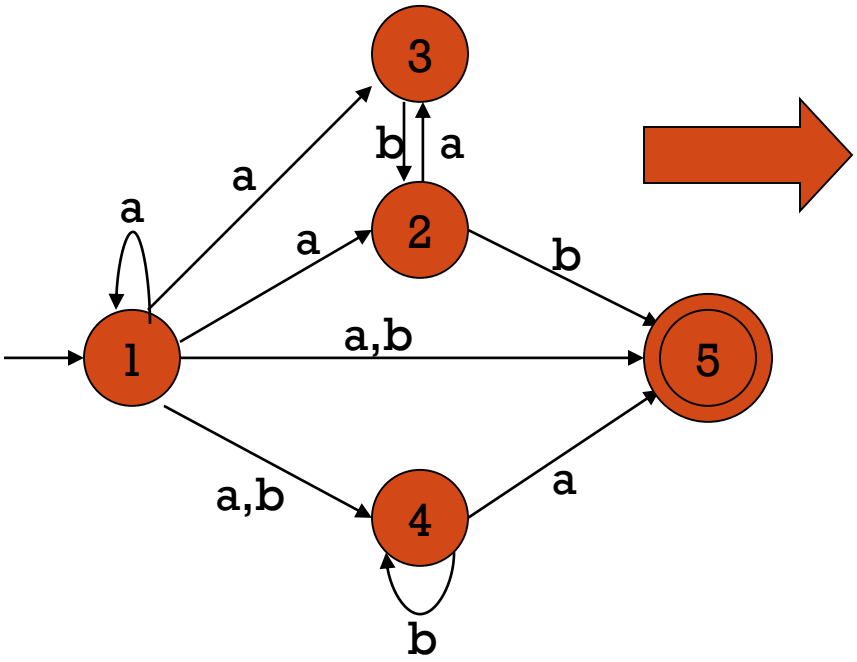
- We can write tokens types as regular expressions
- We have converted Res into NFAs
- We want to convert a NFA into DFA
- The DFA can be mapped into a Transition Table

CONVERT NFA TO DFA

- **Method (Subset Construction)**
- **Input – An NDFA**
- **Output – An equivalent DFA**
  - **Step 1** – Create state table from the given NDFA.
  - **Step 2** – Create a blank state table under possible input alphabets for the equivalent DFA.
  - **Step 3** – Mark the start state of the DFA by q0 (Same as the NDFA).
  - **Step 4** – Find out the combination of States $\{Q_0, Q_1, \ldots, Q_n\}$ for each possible input alphabet.
  - **Step 5** – Each time we generate a new DFA state under the input alphabet columns, we have to apply step 4 again, otherwise go to step 6.
  - **Step 6** – The states which contain any of the final states of the NDFA are the final states of the equivalent DFA.

SUBSET CONSTRUCTION METHOD

NFA without Є-moves



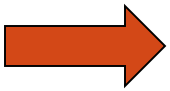
Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	3	5
3	Ø	2
4	5	4
5	Ø	Ø

Subset Construction Method

Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅



Step 1

The set of states resulting from every transition function constitutes a new state. Calculate all reachable states for every such state for every input signal.

Step 2

Repeat this process (step1) until no more new states are reachable.

Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with
Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}

Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}		
{4,5}		

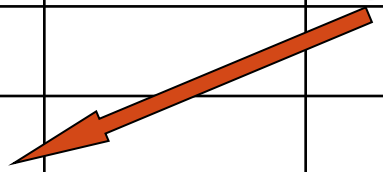
Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}		
{2,4,5}		



Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}		
5		
4		

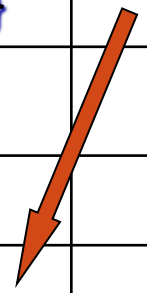
Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5		
4		
{3,5}		



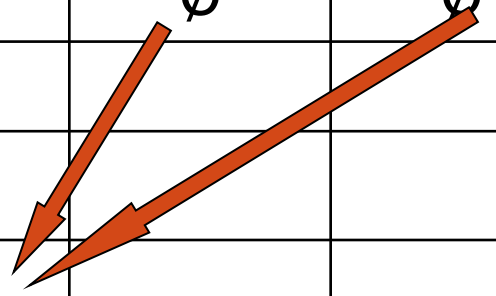
Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	∅	∅
4		
{3,5}		
∅		



Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with
Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	∅	∅
4	5	4
{3,5}		
∅		

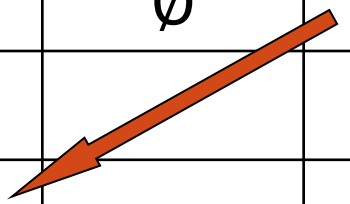
Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	Ø	{2}
4	{5}	{4}
5	Ø	Ø

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	Ø	Ø
4	5	4
{3,5}	Ø	2
Ø		
2		



Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	∅	∅
4	5	4
{3,5}	∅	2
∅	∅	∅
2	3	5

Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	∅	∅
4	5	4
{3,5}	∅	2
∅	∅	∅
2	3	5
3		

Transition table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
2	{3}	{5}
3	∅	{2}
4	{5}	{4}
5	∅	∅

Starts with Initial state

Subset Construction table

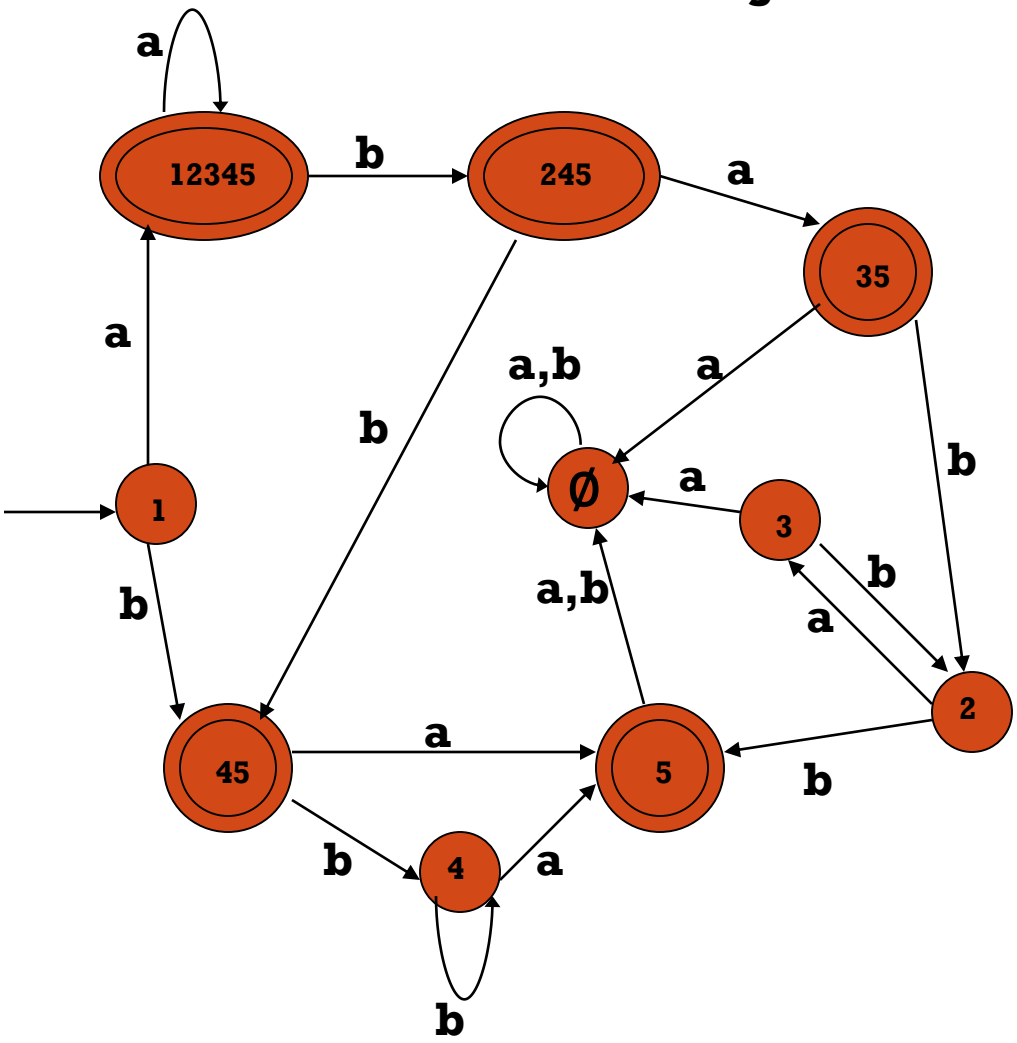
q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	∅	∅
4	5	4
{3,5}	∅	2
∅	∅	∅
2	3	5
3	∅	2

Stops here as there are no more reachable states

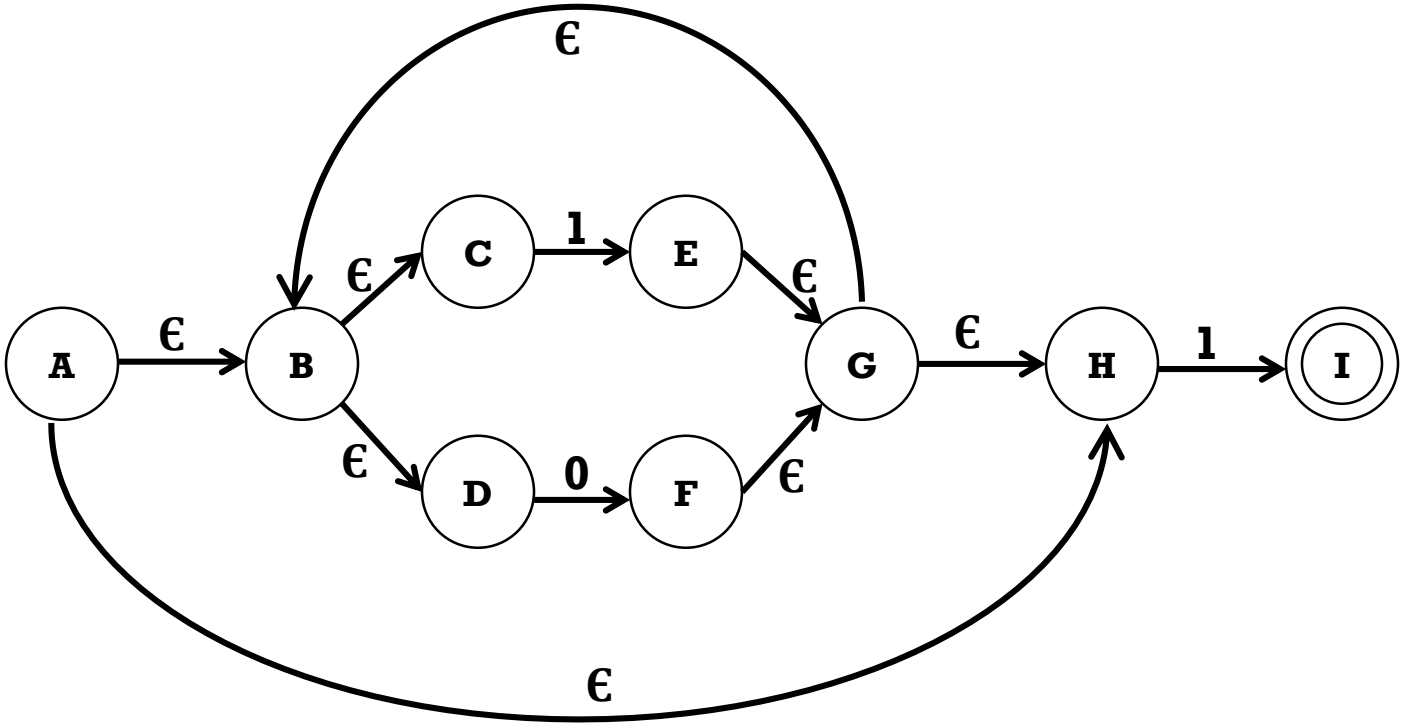
Subset Construction table

q	δ(q,a)	δ(q,b)
1	{1,2,3,4,5}	{4,5}
{1,2,3,4,5}	{1,2,3,4,5}	{2,4,5}
{4,5}	5	4
{2,4,5}	{3,5}	{4,5}
5	∅	∅
4	5	4
{3,5}	∅	2
∅	∅	∅
2	3	5
3	∅	2

Resulting DFA after applying Subset Construction Algorithm



NFA with Є-moves



IF the NFA contains Є-moves we must compute Є-closure(q) which is the set of states reachable from the state q on Є–transition.

Example:

Є-closure (A) = {A,B,C,D,H}

Є-closure (G) = {G,B,C,D,H}

Steps to Convert NFA with ε-move to DFA

Step 1 : Take ∈ closure for the beginning state of NFA as beginning state of DFA.

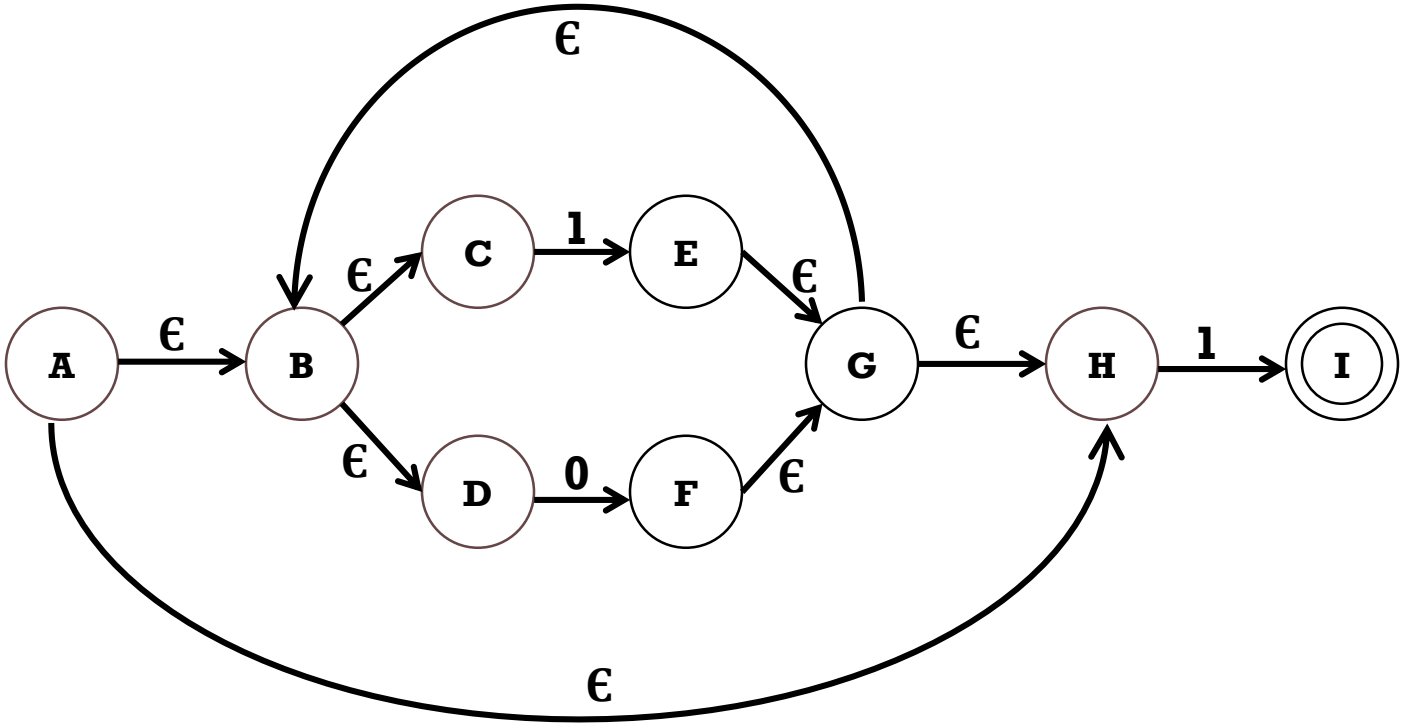
Step 2 : Find the states that can be traversed from the present for each input symbol (union of transition value and their closures for each states of NFA present in current state of DFA).

Step 3 : If any new state is found take it as current state and repeat step 2.

Step 4 : Do repeat Step 2 and Step 3 until no new state present in DFA transition table.

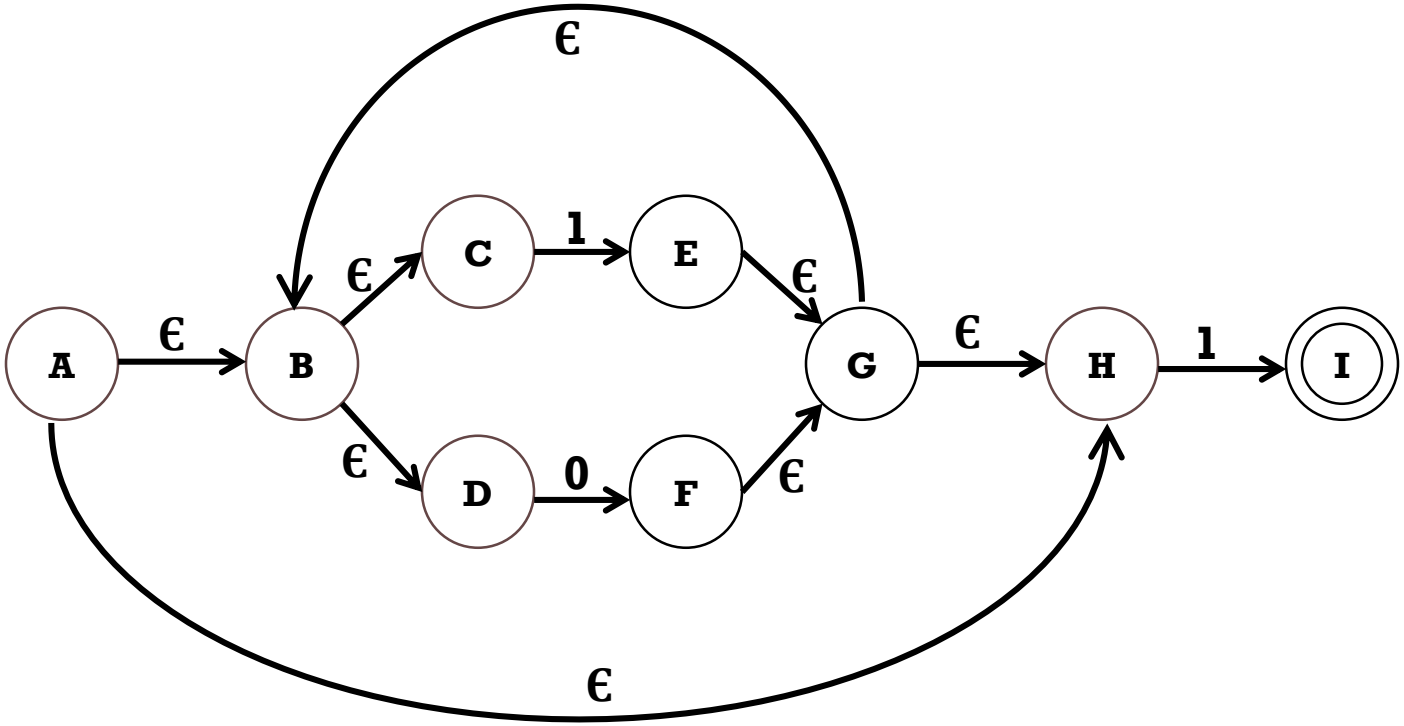
Step 5 : Mark the states of DFA which contains final state of NFA as final states of DFA.

Convert NFA with Є-moves to DFA



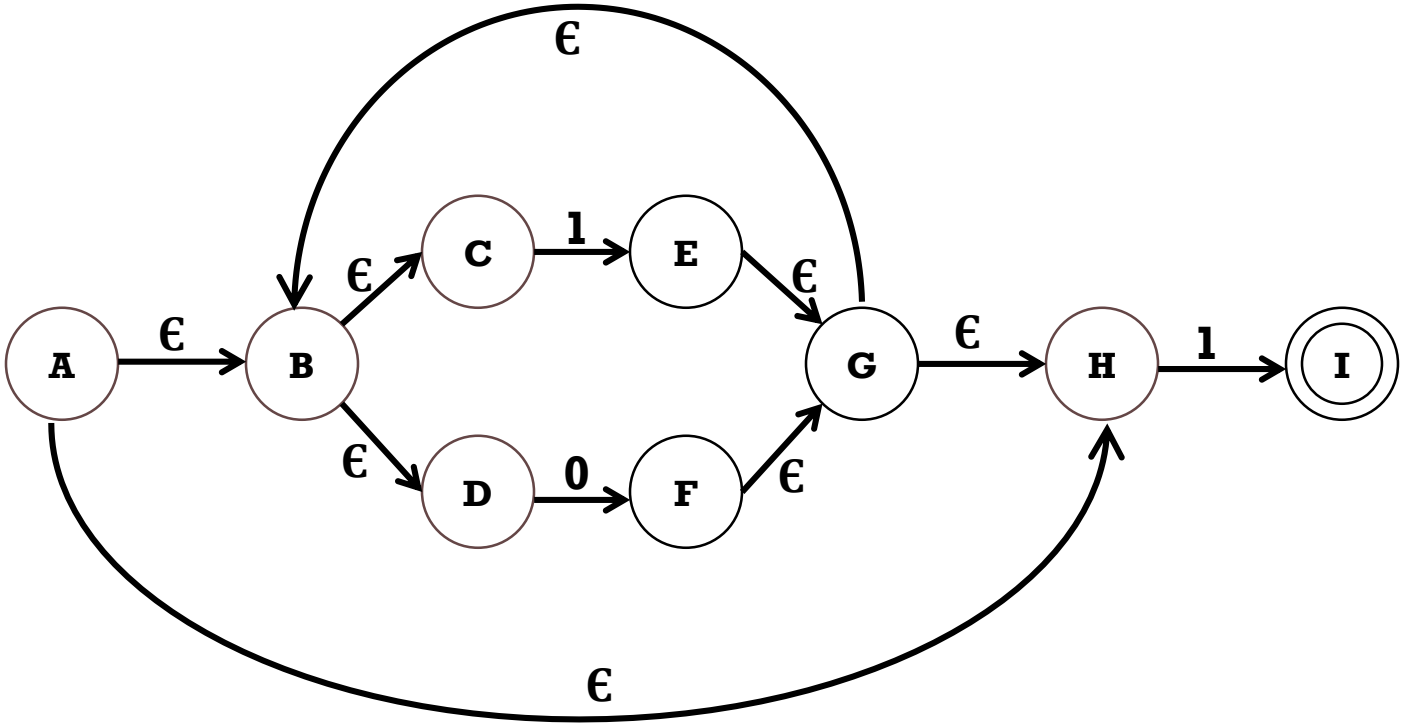
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}		

Convert NFA with Є-moves to DFA



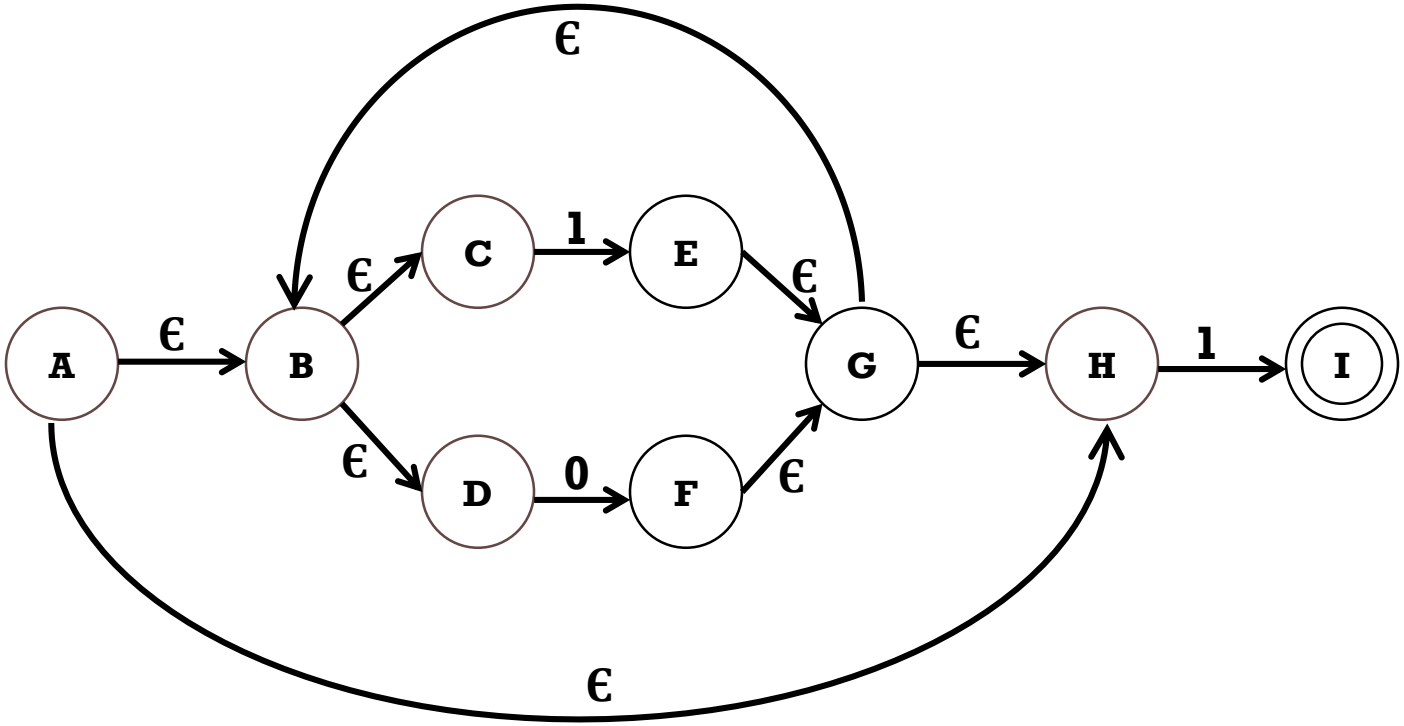
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	

Convert NFA with Є-moves to DFA



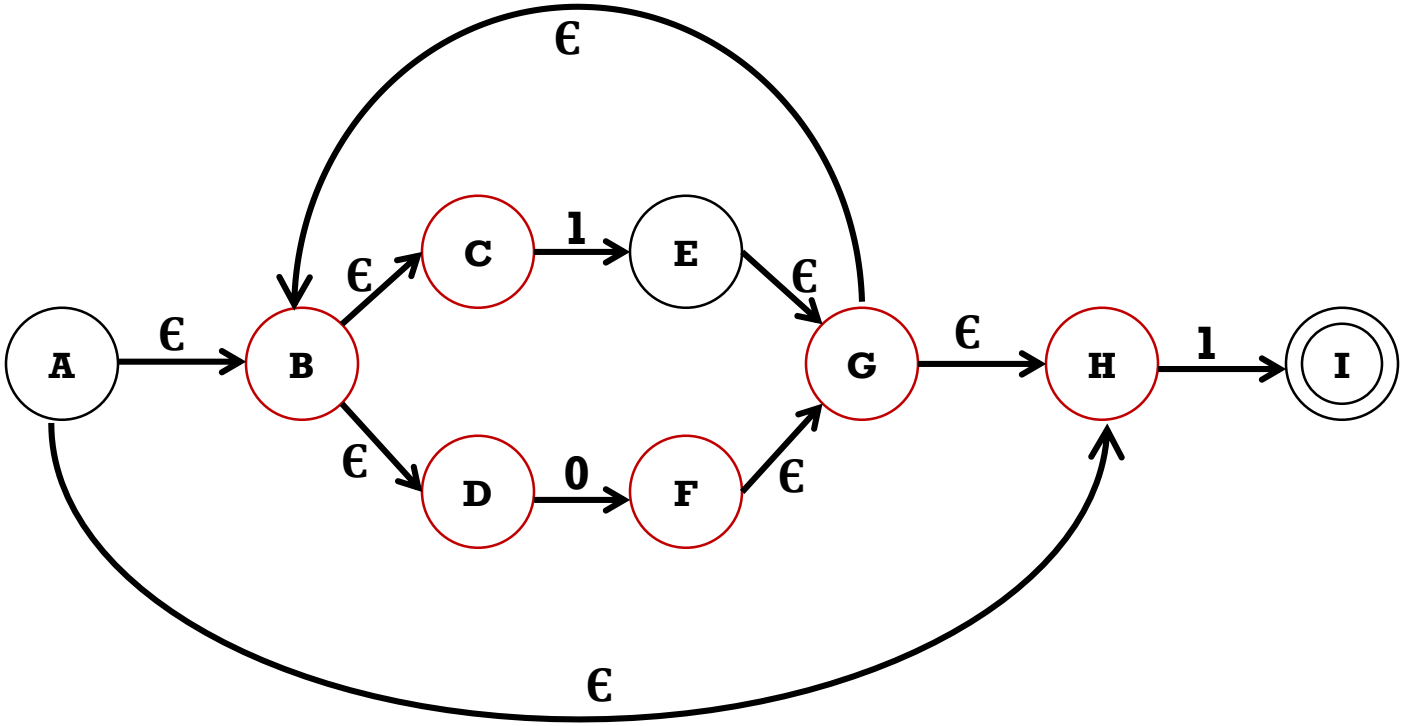
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	{EGHIBCD}

Convert NFA with Є-moves to DFA



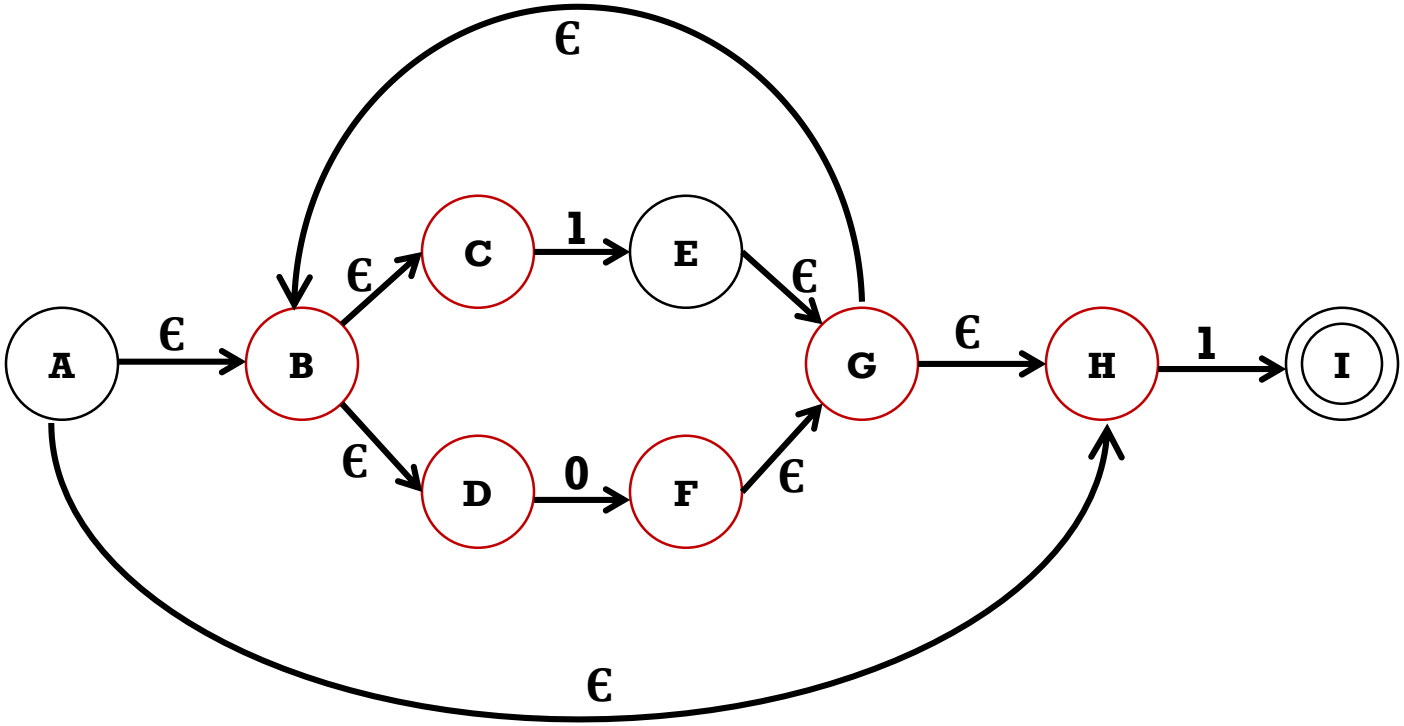
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	{EGHIBCD}
{FGHBCD}		
{EGHIBCD}		

Convert NFA with Є-moves to DFA



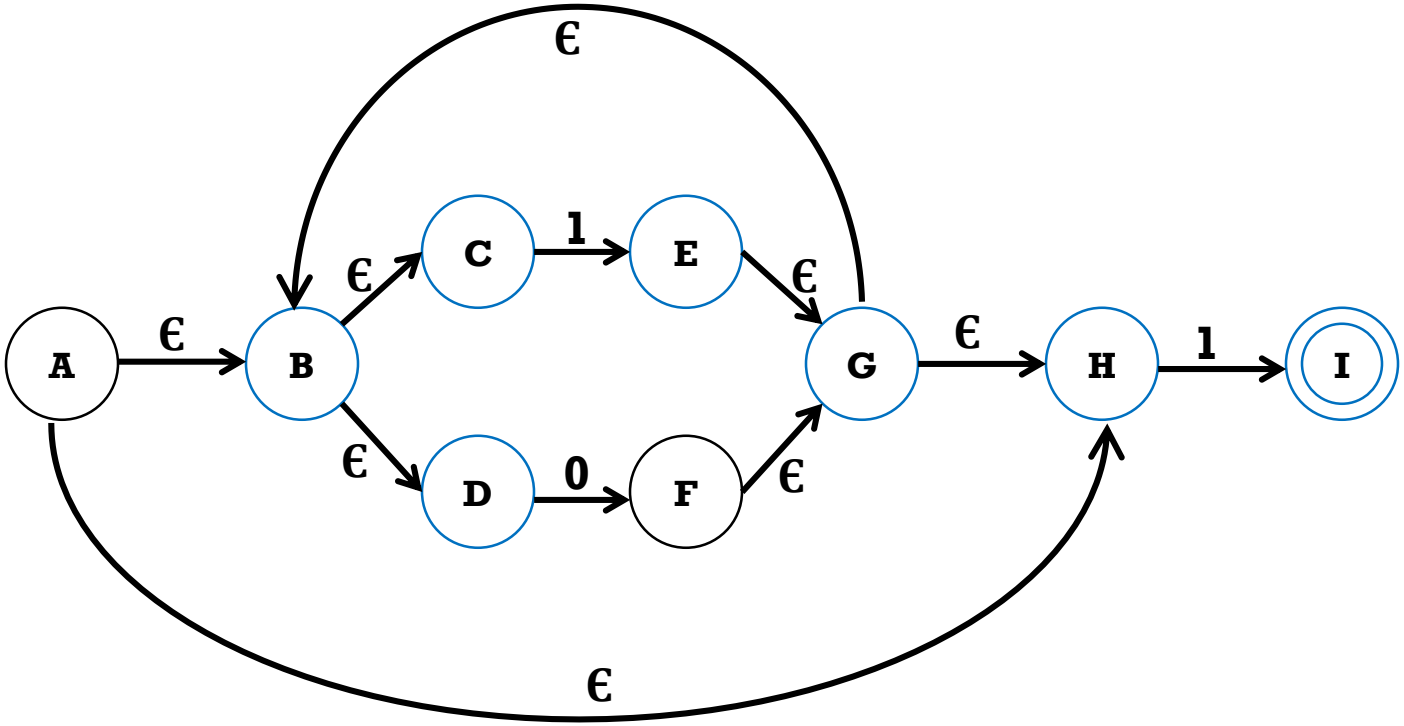
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	{EGHIBCD}
{FGHBCD}		
{EGHIBCD}		

Convert NFA with Є-moves to DFA



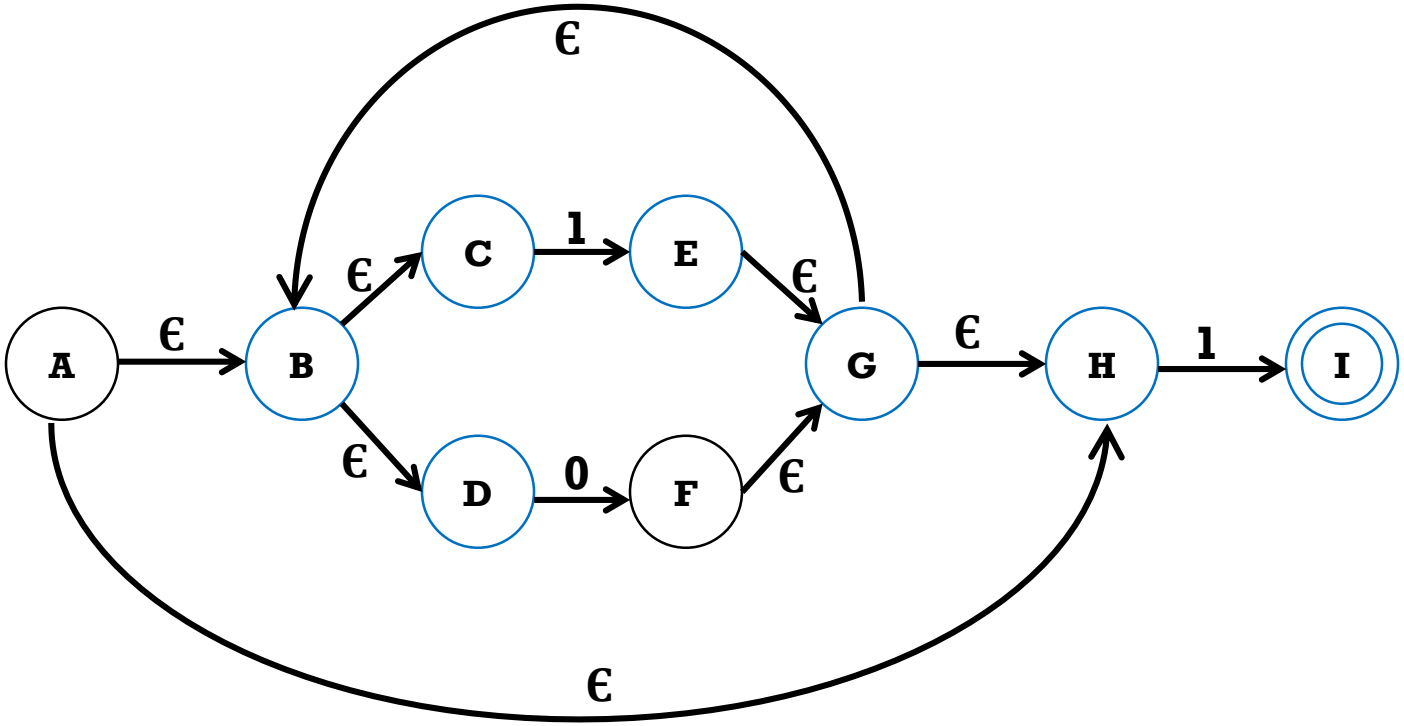
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	{EGHIBCD}
{FGHBCD}	{FGHBCD}	{EGHIBCD}
{EGHIBCD}		

Convert NFA with Є-moves to DFA



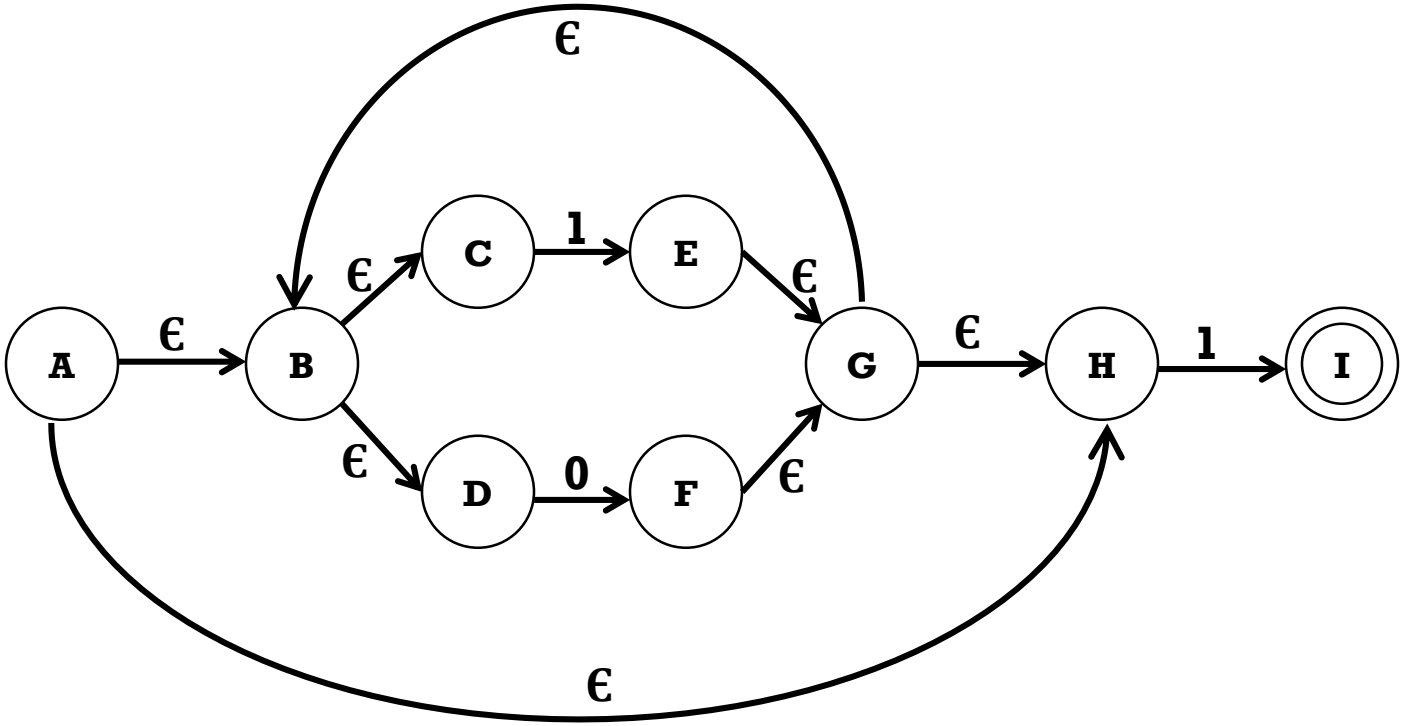
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	{EGHIBCD}
{FGHBCD}	{FGHBCD}	{EGHIBCD}
{EGHIBCD}		

Convert NFA with Є-moves to DFA



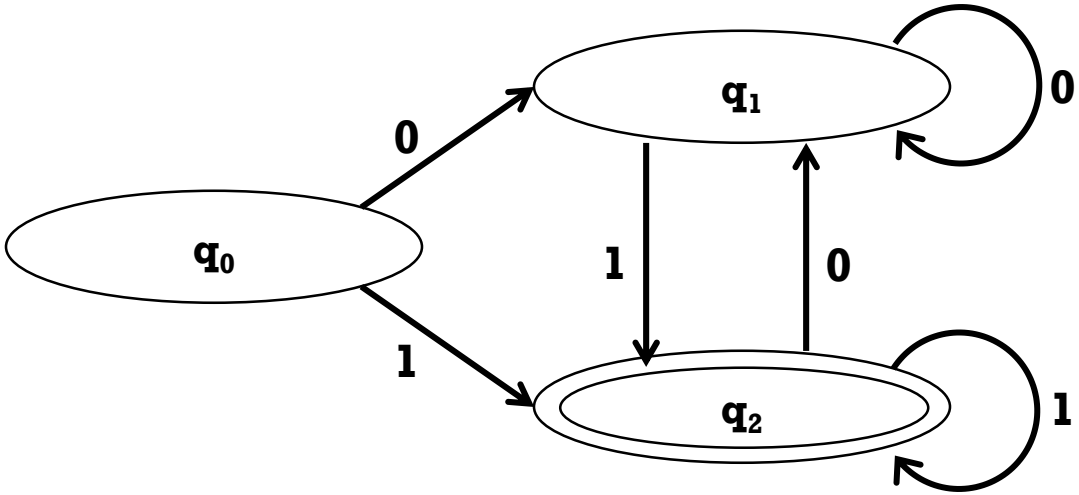
q	*δ(q,0)*	*δ(q,1)*
{ABCDH}	{FGHBCD}	{EGHIBCD}
{FGHBCD}	{FGHBCD}	{EGHIBCD}
{EGHIBCD}	{FGHBCD}	{EGHIBCD}

Convert NFA with Є-moves to DFA



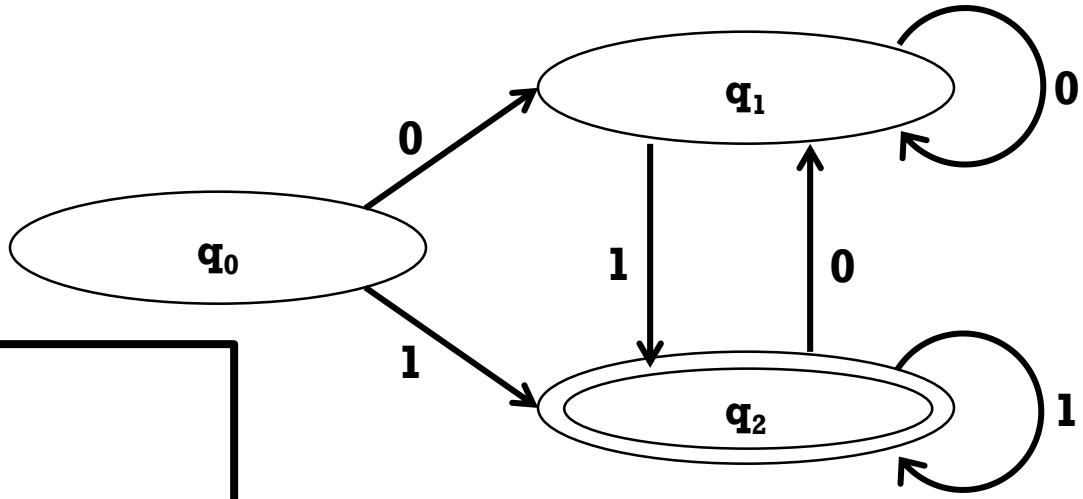
q	*δ(q,0)*	*δ(q,1)*
q_0	q_1	q_1
q_1	q_1	q_2
q_2	q_2	q_2

The Resulting DFA



q	$\delta(q,0)$	$\delta(q,1)$
q_0	q_1	q_1
q_1	q_1	q_2
q_2	q_2	q_2

Implementing DFA




```
q = q0
c = nextInputChar();
while ( c != eof ){
        q = δ(q, c);
        c = nextInputChar();
}
if( q ∈ F )
        return "Accept";
else
        return "Reject";
```

q	*δ(q,a)*	*δ(q,b)*
q_0	q_1	q_1
q_1	q_1	q_2
q_2	q_2	q_2

Assignment 3

Construct the DFA equivalent to the following NFAs