Every DFA is an NFA, but not vice versa

But there is an equivalent DFA for every NFA

DFA

$\delta = Q \times \Sigma \rightarrow Q$

NFA

$\delta = Q \times \Sigma \rightarrow 2^Q$

$NFA \cong DFA$

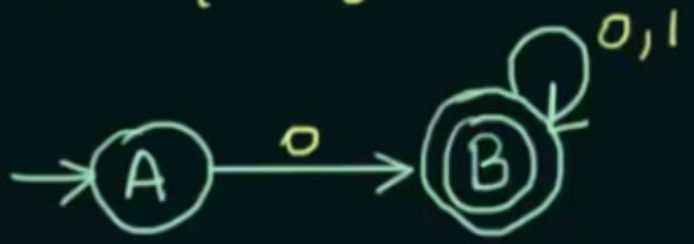
L = { Set of all strings over (0,1) that starts with '0' }

NFA ≅ DFA

L = { Set of all strings over (0,1) that starts with '0' }

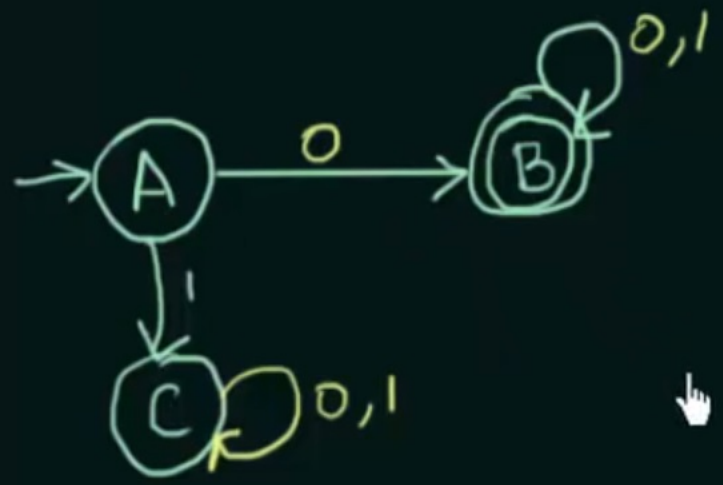
$\Sigma = \{0, 1\}$

NFA



	0	1
A	B	ϕ
B	B	B

DFA



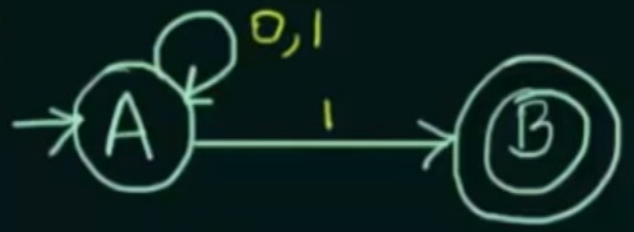
	0	1
A	B	C
B	B	B
C	C	C

C - Dead State / Trap State

L = { Set of all strings over (0,1) that ends with '1' }

$\Sigma = 0, 1$

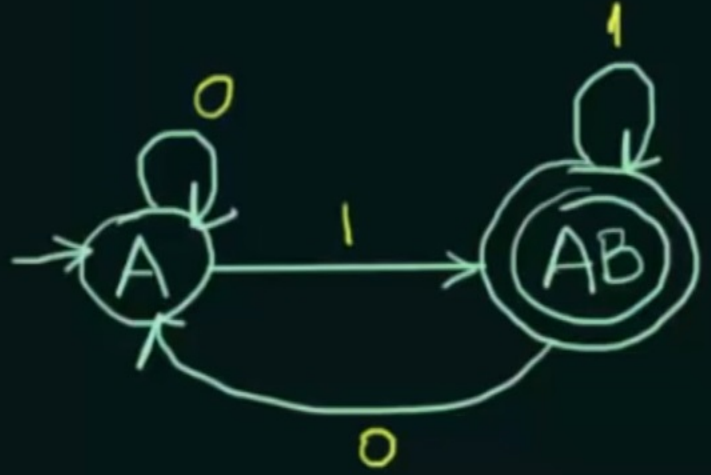
NFA



	0	1
A	{A}	{A, B}
B	ϕ	ϕ

Subset construction method

DFA

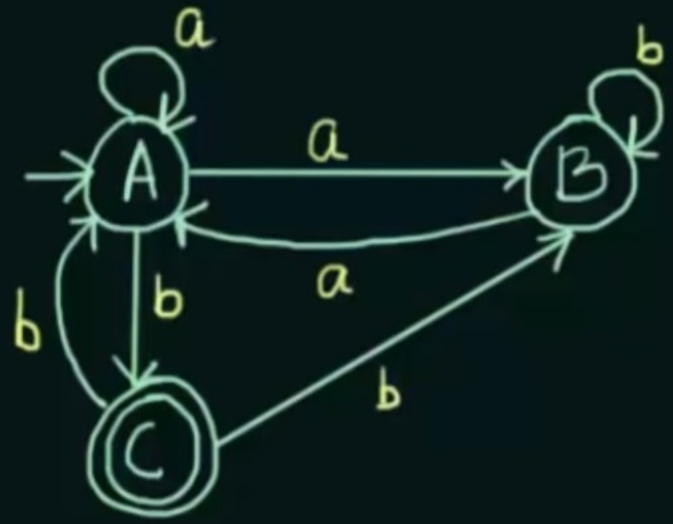


	0	1
A	{A}	{AB}
AB	{A}	{AB}

AB - Single State

Find the equivalent DFA for the NFA given by $M = [\{A,B,C\}, (a,b), \delta, A, \{C\}]$ where δ is given by:

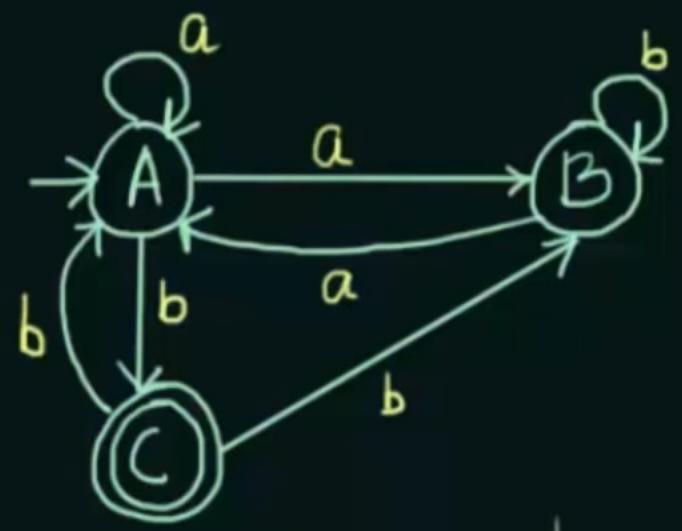
	a	b
→ A	A, B	C
B	A	B
(C)	–	A, B



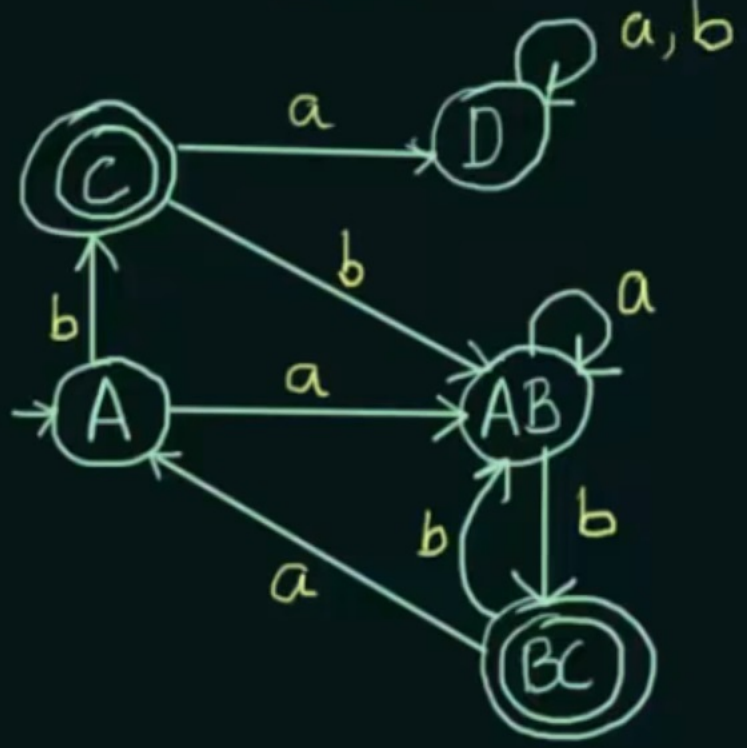
	a	b
→ A	AB	C
AB	AB	BC
BC	A	AB

Find the equivalent DFA for the NFA given by $M = [\{A,B,C\}, \{a,b\}, \delta, A, \{C\}]$ where δ is given by:

	a	b
→A	A,B	C
B	A	B
(C)	–	A,B



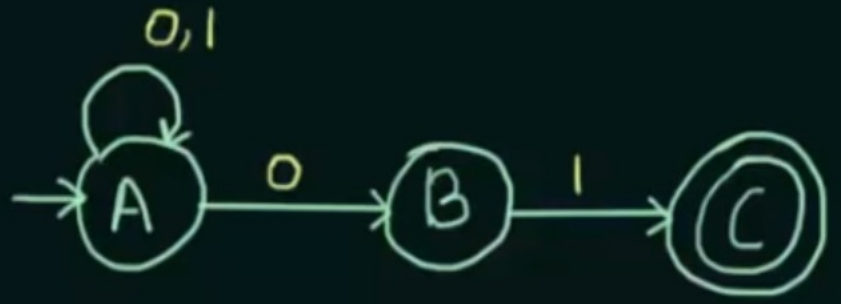
	a	b
→ A	AB	C
AB	AB	BC
(BC)	A	AB
(C)	D	AB
D	D	D



Given below is the NFA for a language

L = { Set of all strings over (0,1) that ends with '01' }. Construct its equivalent DFA

NFA



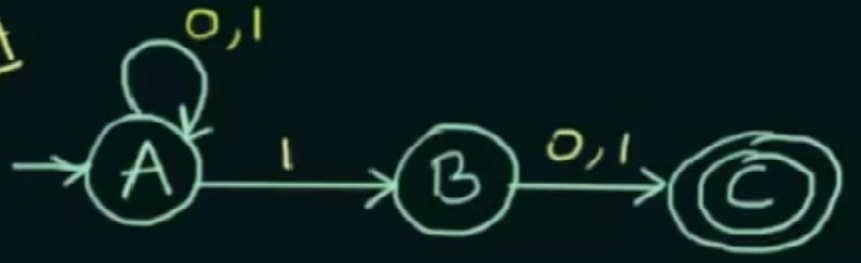
	0	1
→A	A, B	A
B	ϕ	C
(C)	ϕ	ϕ

	0	1
→A	AB	A
AB	AB	AC
(AC)	AB	A

Design an NFA for a language that accepts all strings over {0,1} in which the second last symbol is always '1'. Then convert it to its equivalent DFA.

Eg. 1010
1110
1101010

NFA



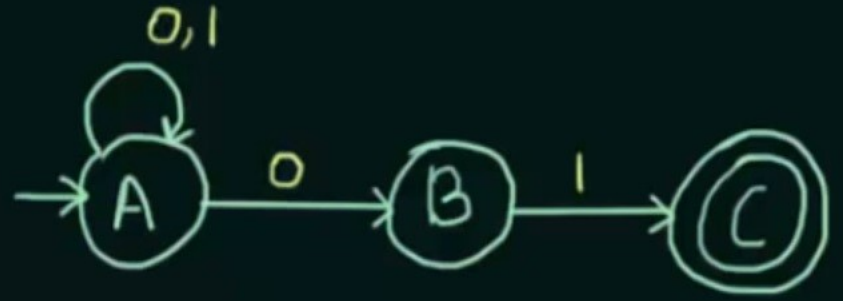
	0	1
→A	A	A,B
B	C	C
(C)	ϕ	ϕ

DFA

Given below is the NFA for a language

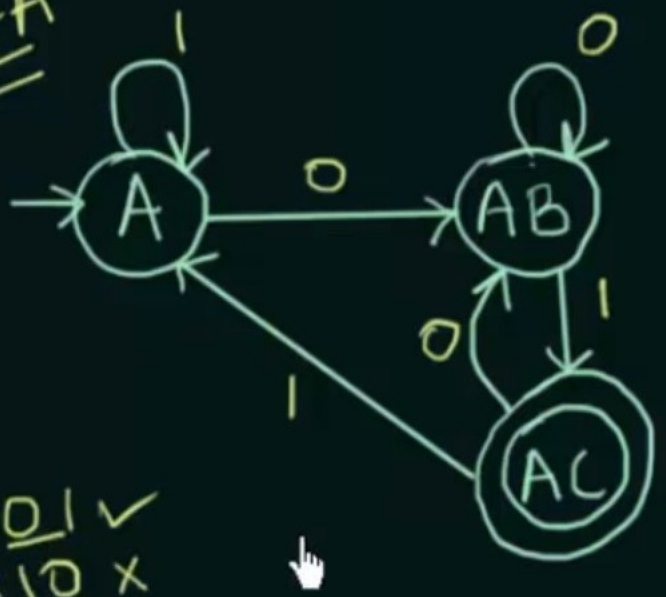
L = { Set of all strings over (0,1) that ends with '01' }. Construct its equivalent DFA

NFA



	0	1
→A	A, B	A
B	ϕ	C
(C)	ϕ	ϕ

DFA

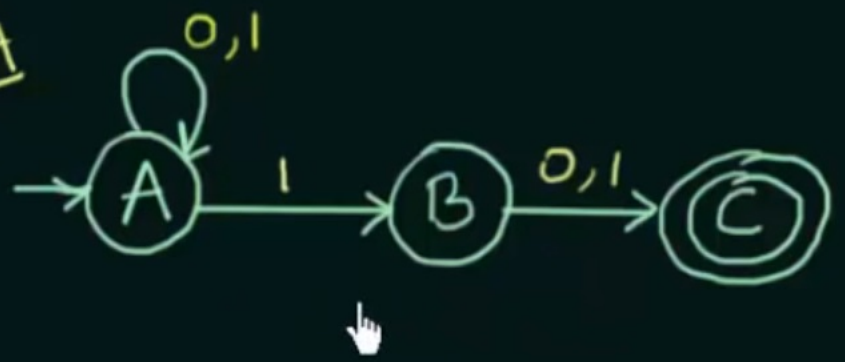


	0	1
→A	AB	A
AB	AB	AC
(AC)	AB	A

Second last symbol is always '1'. Then convert it to its equivalent DFA.

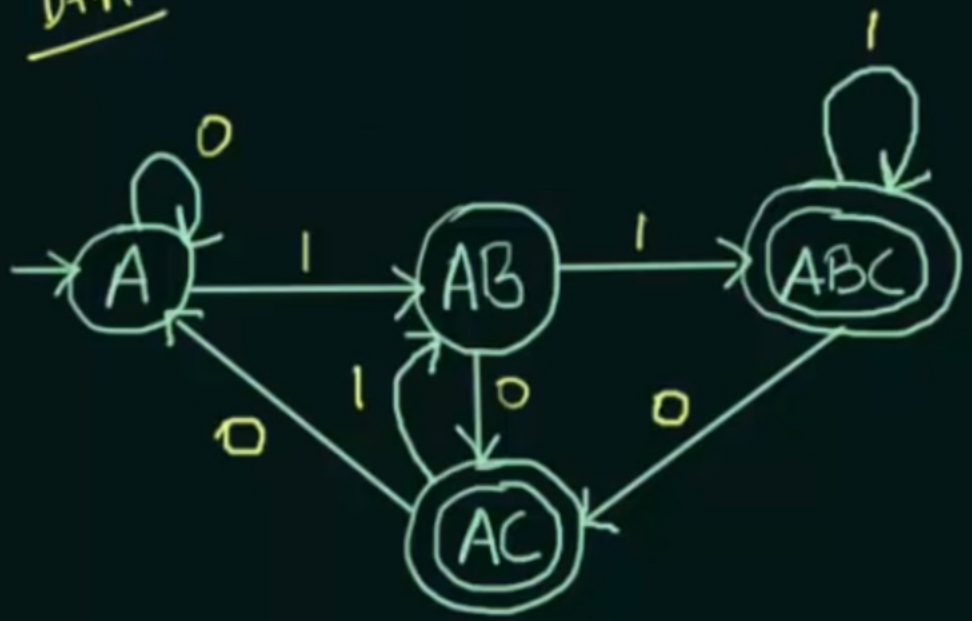
Eg. 1010 ✓
110 ✓
1101010 ✓

NFA



	0	1
→A	A	A,B
B	C	C
(C)	ϕ	ϕ

DFA



	0	1
→A	A	AB
AB	AC	ABC
(AC)	A	AB
(ABC)	AC	ABC