Morning

Regular Expression

1. Definition
2. Designing
3. Equivalence with FA

Operation of Languages

L={ 0,11 }, M ={ ε , 001 }

- Union

L$\cup$M={ 0, 11 , ε , 001 }

- Concatenation

LM={ 0, 11, 0001 , 11001 }

ML={ 0, 11, 0010, 00111 }

Operation of Languages

$L = \{0, 11\}$

- Star (Closure , Kleene Closure)

$L^0 = \{\varepsilon\}$, $L^1 = \{0, 11\}$, $L^2 = \{00, 011, 110, 1111\}$

$L^3 = \{000, 0011, 0110, 1100, 01111, 11011, 11110, 111111\}$

$$L^* = \bigcup_{i=0}^{\infty} L^i$$

$\phi^* = ?$

Arithmetical Expression

$$0,\ 1+2,\ 3\times(5-2),\ (56-7)^2\ ,\ \ldots\ldots$$

- Formal definition
- Inductive definition
  - Any number is a arithmetical expression
  - If **a** and **b** are arithmetical expressions , then

    so is $a+b, a-b, a\div b, a\times b, a^n, (a)$.

Building Regular Expressions

BASIS

1. ε is a regular expression, denoting the languages $\{\varepsilon\}$.

2. ϕ is a regular expression, denoting the languages ϕ.

3. For each a in Σ, a is a regular expression and denotes the language $\{a\}$.

Building Regular Expressions

INDUCTION

1. If E and F are regular expressions, denoting the language $L(E)$ and $L(F)$, then $E+F$, EF and E^* are regular expressions that denote the languages $L(E) \cup L(F)$, $L(E)L(F)$ and $(L(E))^*$.

2. If E is a RE then so is (E) .

Example 5.1 What is the language defined by r

$$r = (a+b)^*(a+bb)$$

$a \rightarrow \{ \mathrm{a} \}$, $b \rightarrow \{ \mathrm{b} \}$

$a+b \rightarrow \{ \mathrm{a} \} \cup \{ \mathrm{b} \} = \{\mathrm{a}, \mathrm{b}\}$

$bb \rightarrow \{ \mathrm{b} \}\{ \mathrm{b} \} = \{ \mathrm{bb} \}$

$a+bb \rightarrow \{ \mathrm{a} \} \cup \{ \mathrm{bb} \} = \{ \mathrm{a}, \mathrm{bb} \}$

$(a+b)^* \rightarrow \{ \mathrm{a}, \mathrm{b} \}^*$

$(a+b)^*(a+bb) \rightarrow \{ \mathrm{a}, \mathrm{b} \}^*\{ \mathrm{a}, \mathrm{bb} \}$

$L(r) = \{ a, bb, aa, abb, ba, bbb, \ldots\ldots \}$

Example 5.2 What is the language defined by r

$$r = (aa)^*(bb)^*b$$

$$L(r) = (\{a\}\{a\})^*(\{b\}\{b\})^*\{b\}$$

$$= (\{aa\})^*(\{bb\})^*\{b\}$$

$$= \{aa\}^* \{bb\}^*\{b\}$$

$$= \left\{ a^{2n}b^{2m+1} \mid n \geq 0, m \geq 0 \right\}$$

Example 5.3

Write a regular expression for the set of strings that consist of alternating 0's and 1's.

Partition :

$010101\ldots0101 \longrightarrow (01)^*$

$101010\ldots1010 \longrightarrow (10)^*$

$0101010\ldots1010 \longrightarrow 0(10)^*$ or $(01)^*0$

$101010\ldots10101 \longrightarrow (10)^*1$ or $1(01)^*$

The regular expression :

$$(01)^*+(10)^*+0(10)^*+(10)^*1 \Rightarrow (\varepsilon+0)(10)^*(\varepsilon+1)$$

Example 5.4 Design regular expression for L

$L=\{w \mid w \in \{0, 1\}^*$ and w contains 01 $\}$

$r(L) = (0+1)^*01(0+1)^*$

Example 5.5 Design regular expression for L

$L=\{w \mid w$ consists of 0's and 1's , and the third symbol from the right end is 1 $\}$

$r(L) = (0+1)^*1(0+1)(0+1)$

Example 5.6 Design regular expression for L

$L = \{ w \mid w \in \{0, 1\}^*$ and w has no pair of consecutive 0's $\}$

Partition :

no 0 ⟶ 1^*

one 0 ⟶ 1^*01^*

more 0's ⟶ $1^*(011^*)^*(0+\varepsilon)$

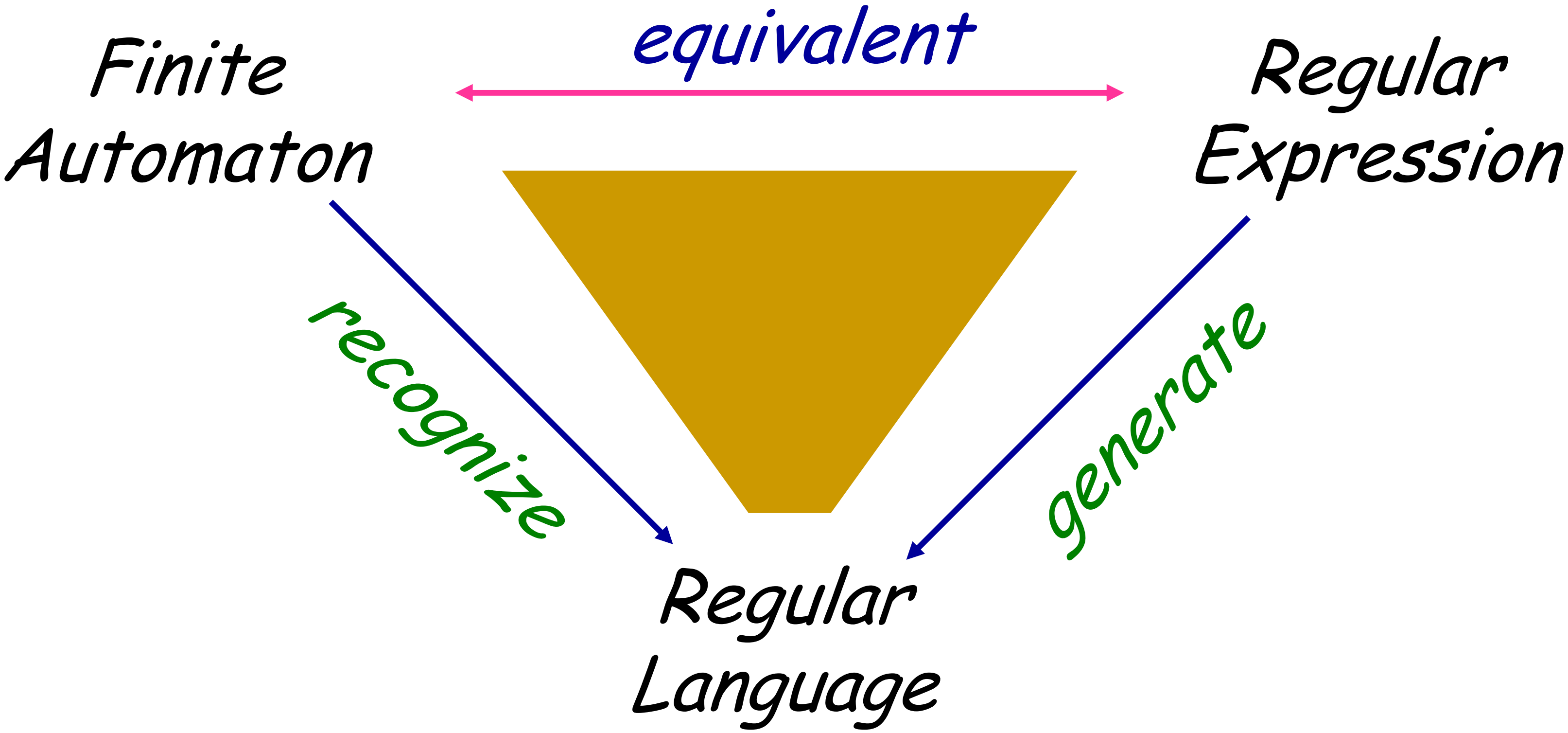
Example 5.6 Design regular expression for L

$L = \{ w \mid w \in \{0, 1\}^*$ and w has no pair of consecutive 0's $\}$

$$r = (1^*011^*)^*(0+\varepsilon) + 1^*(0+\varepsilon)$$

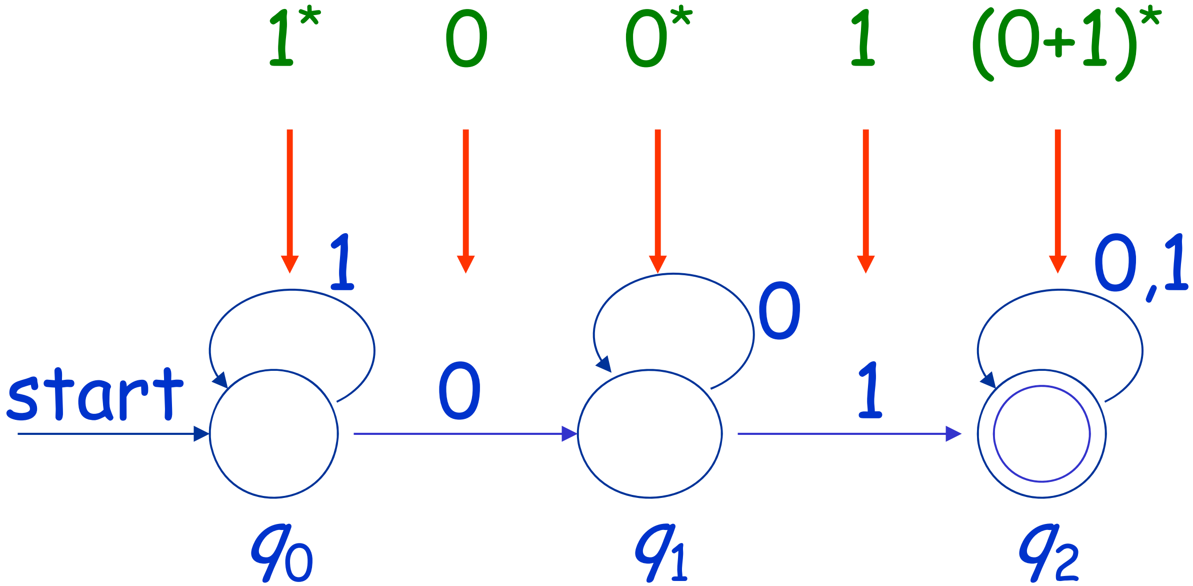
$$r = (1+01)^*(0+\varepsilon)$$

FA & RE



equivalent
Finite Automaton
Regular Expression
recognize
generate
Regular Language

Construct RE from FA

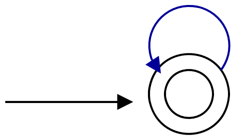


$L = \{ w \mid w \in \{0, 1\}^* \text{ and } w \text{ contains } 01 \}$

RE: $(0+1)^*01(0+1)^* \Rightarrow 1^*00^*1(0+1)^*$

Construct RE from FA

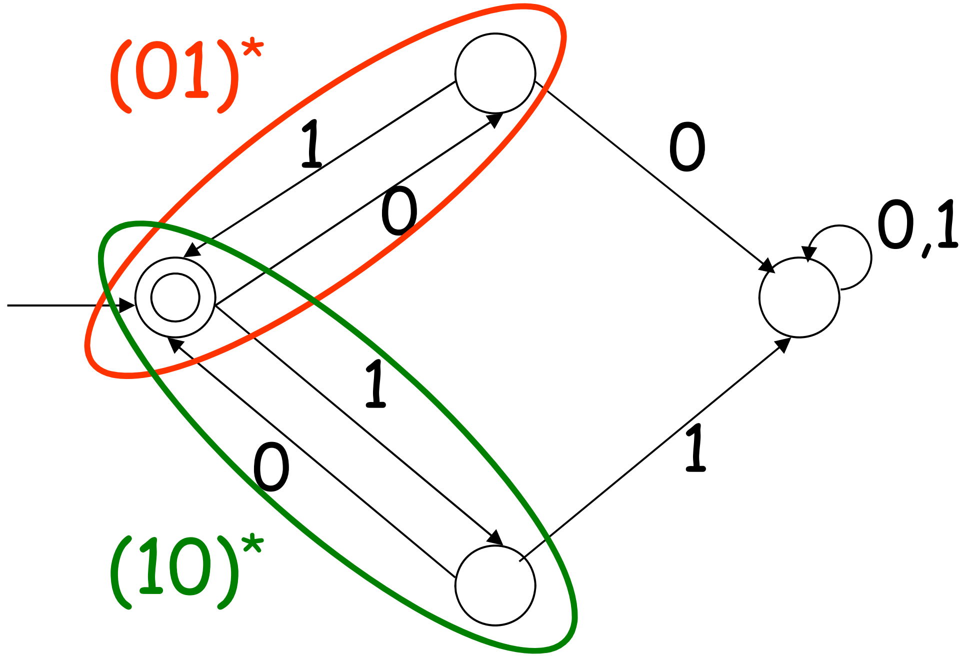
01+10



(just notation)

⇐

(01)*

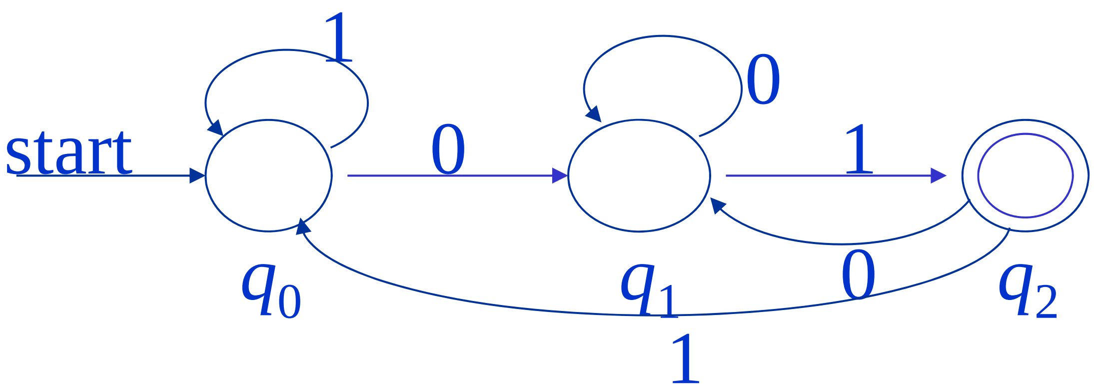


(10)*

(01+10)*

Construct RE from FA

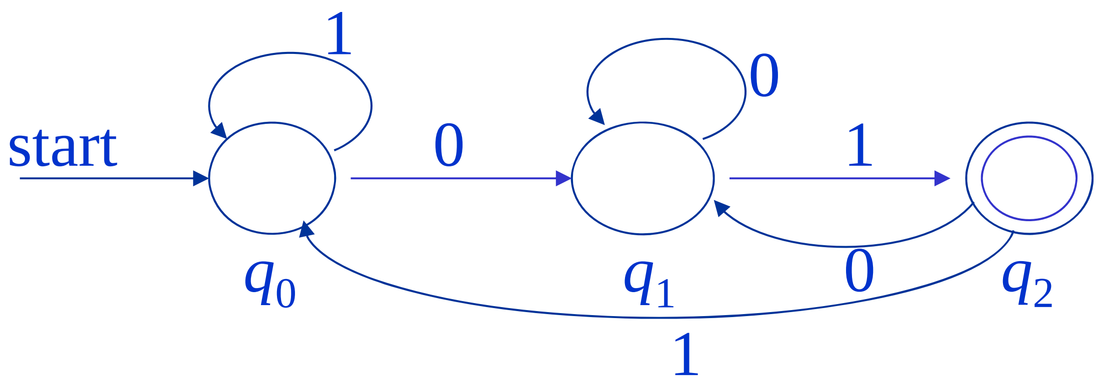
$1^* \quad 0 \quad 0^* \quad 1 \quad (00^*1)^*$



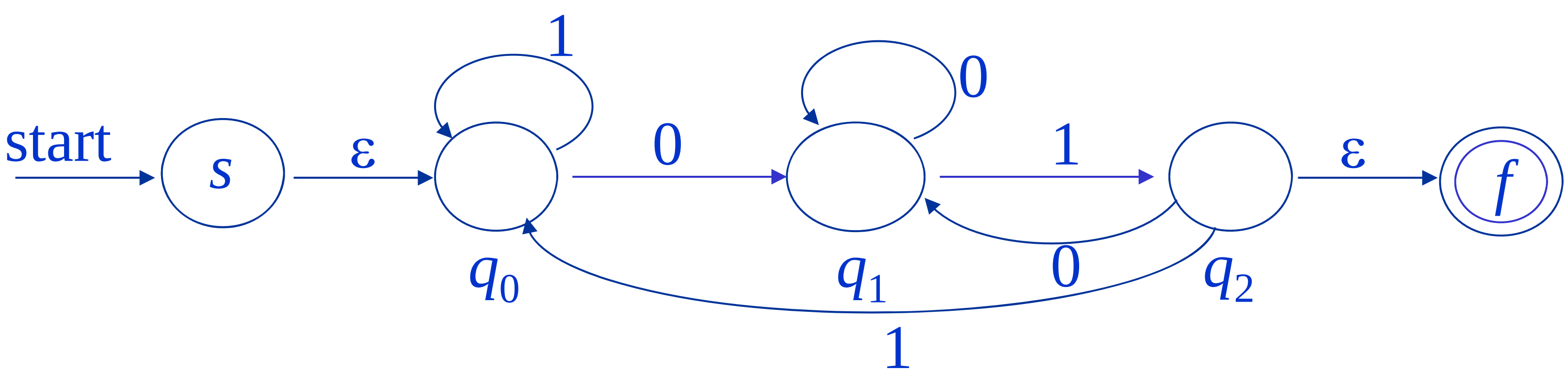
$1^* \quad 0 \quad 0^* \quad 1 \quad (11^*00^*1)^*$

$1^*00^*1\ (00^*1+11^*00^*1)^*$

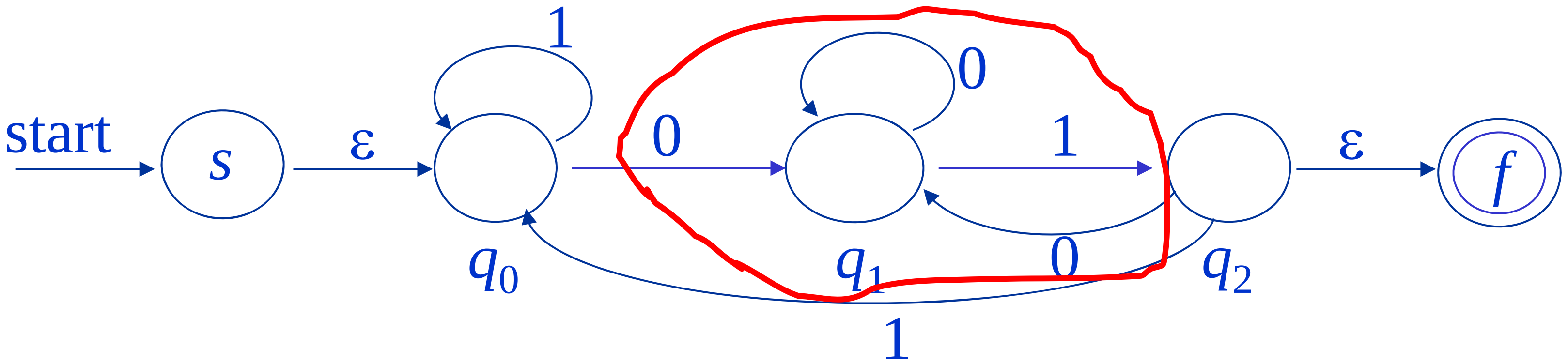
Construct RE by deleting states



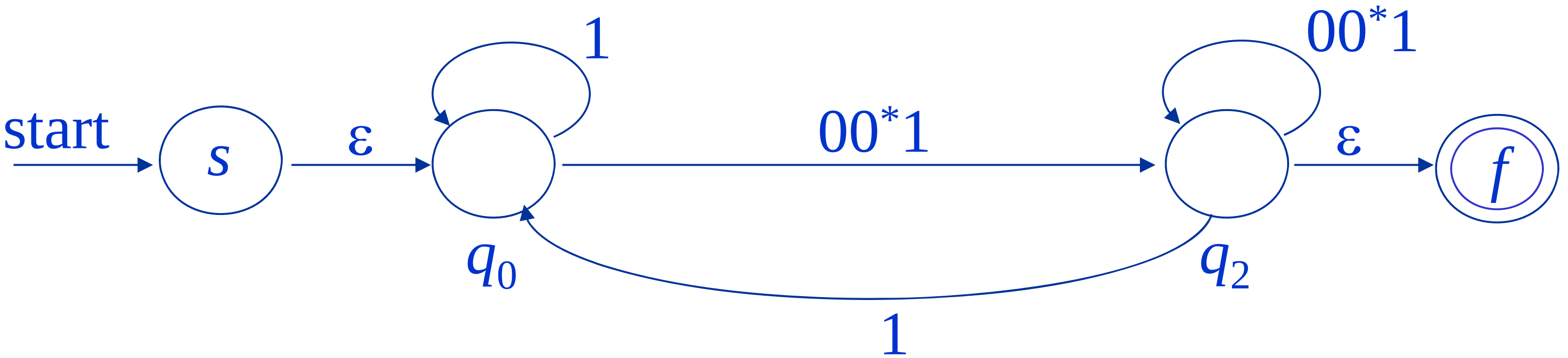
add two states, s and f :



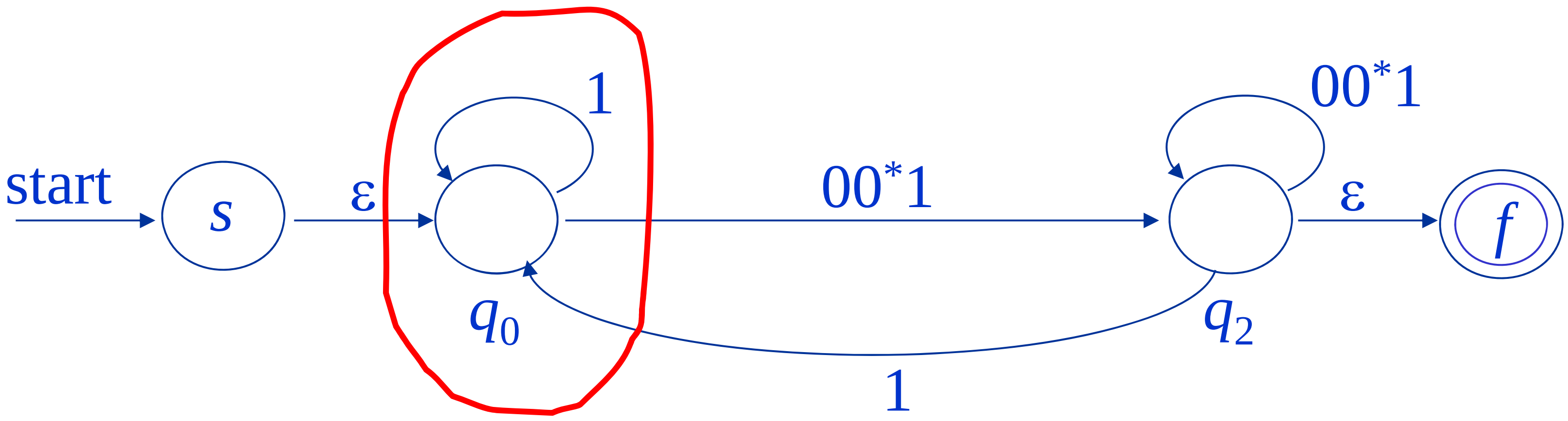
Construct RE by deleting states



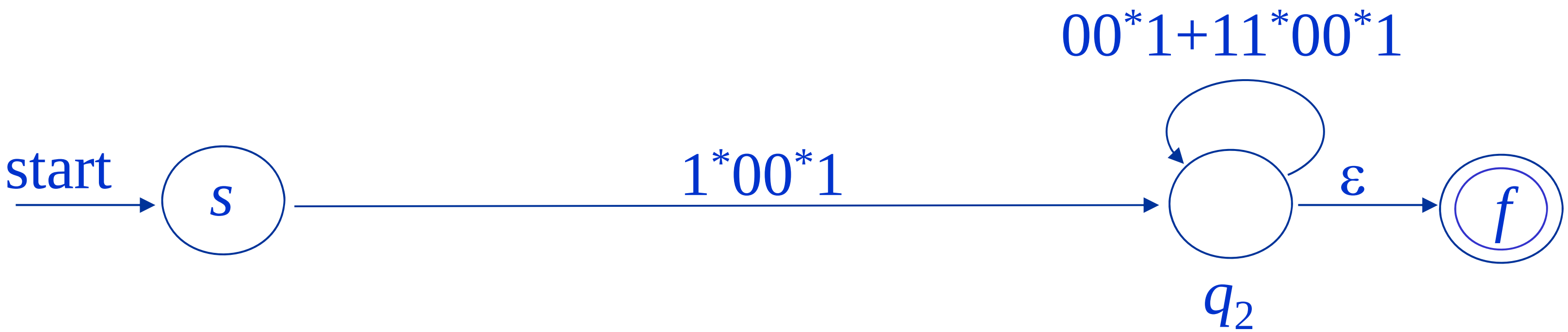
delete q_1 :



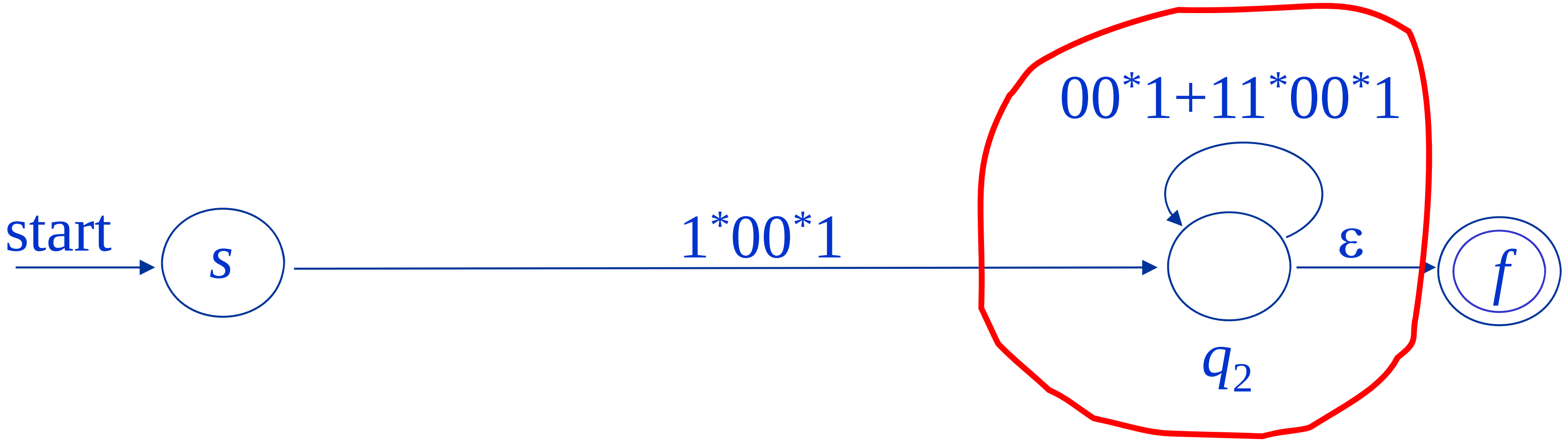
Construct RE by deleting states



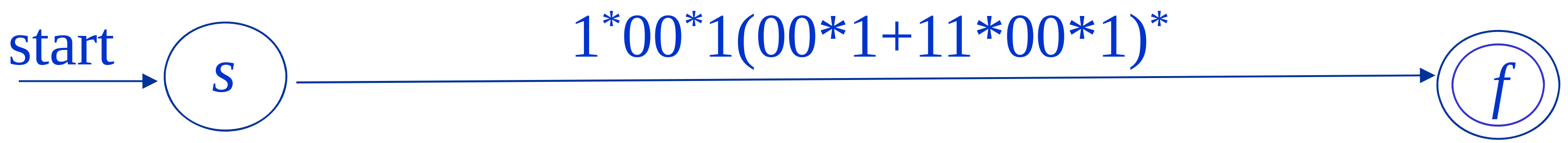
delete q_0 :



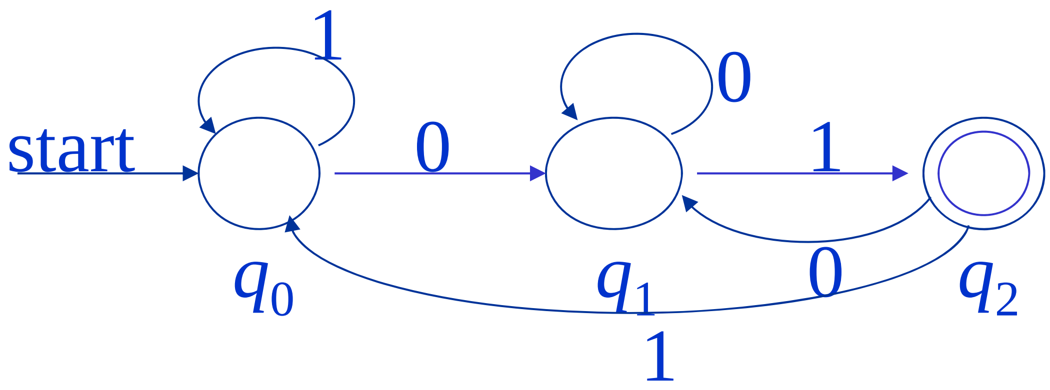
Construct RE by deleting states



delete q_2 :



Construct RE by Induction



- Pick every label on the path from q_0 to q_2

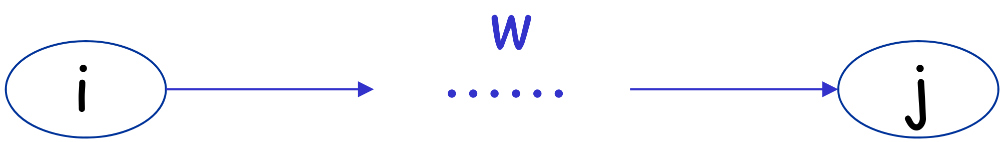
---- one by one

- Form every RE on the path from q_0 to q_2

---- one by one

Construct RE by Induction

- Q={1,2,3,……,n}
- $R_{ij}^{(k)} : 0 \le k \le n$
  - regular expression of path from *i* to *j*
  - no inner node is greater than *k*



$$R_{ij}^{(k)} \Rightarrow w$$

Construct RE by Induction

Basis $k = 0, \; i \neq j$

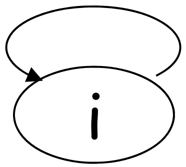
$$i \qquad j \quad \Rightarrow \quad R_{ij}^{(0)} = \phi$$

$$i \xrightarrow{a} j \quad \Rightarrow \quad R_{ij}^{(0)} = a$$

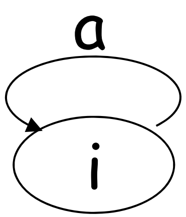
$$i \xrightarrow{a_1, \ldots, a_n} j \quad \Rightarrow \quad R_{ij}^{(0)} = a_1 + a_2 + \cdots + a_n$$

Construct RE by Induction

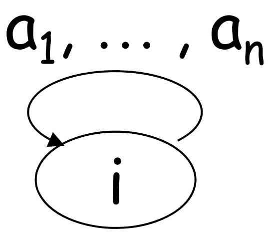
Basis $k = 0$, $i = j$



$$\Rightarrow \quad R_{ij}^{(0)} = \varepsilon + \phi$$



$$\Rightarrow \quad R_{ij}^{(0)} = \varepsilon + a$$

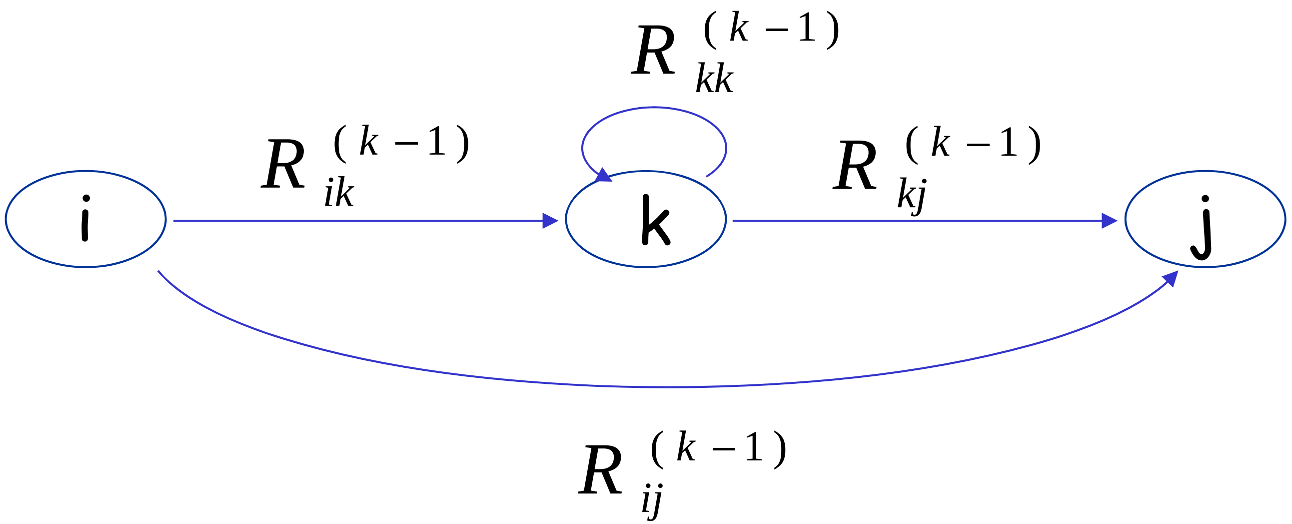


$$\Rightarrow \quad R_{ij}^{(0)} = \varepsilon + a_1 + a_2 + \cdots + a_n$$

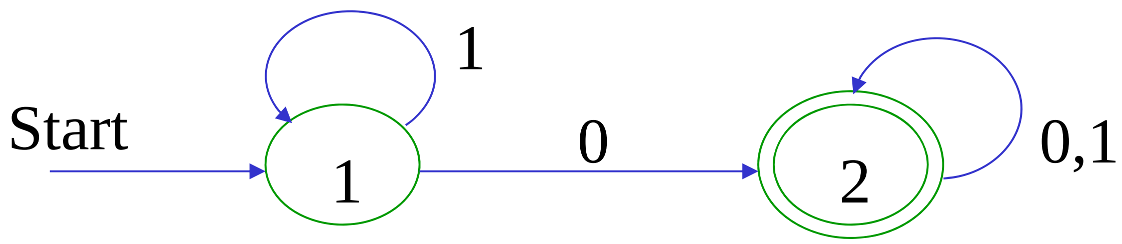
Construct RE by Induction

Induction $k \geq 1$

$$R_{ij}^{(k)} = R_{ij}^{(k-1)} + R_{ik}^{(k-1)} (R_{kk}^{(k-1)})^* R_{kj}^{(k-1)}$$



Example 5.7 Convert FA into regular expression



$$R_{11}^{(0)} = \varepsilon + 1, \quad R_{12}^{(0)} = 0, \quad R_{21}^{(0)} = \phi, \quad R_{22}^{(0)} = \varepsilon + 0 + 1$$

$$R_{11}^{(1)} = 1^*, \quad R_{12}^{(1)} = 1^*0, \quad R_{21}^{(1)} = \phi, \quad R_{22}^{(1)} = \varepsilon + 0 + 1$$

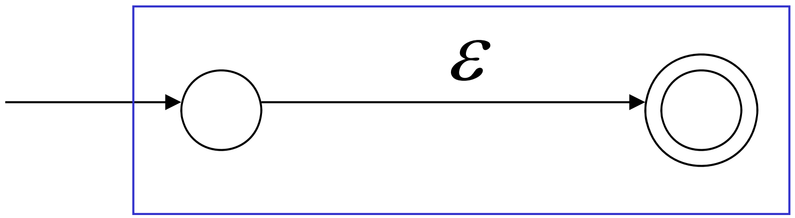
$$R_{11}^{(2)} = 1^*, \quad R_{12}^{(2)} = 1^*0(0+1)^*, \quad R_{21}^{(2)} = \phi, \quad R_{22}^{(2)} = (0+1)^*$$

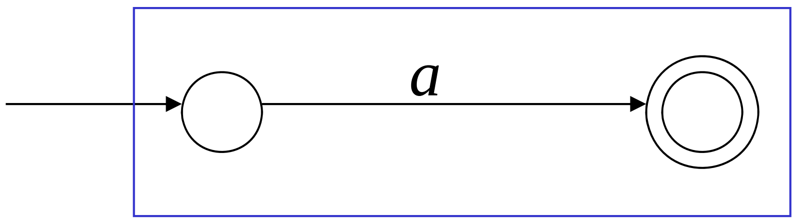
What we need is :

$$R_{12}^{(2)} = 1^*0(0+1)^*$$

Construct FA from RE

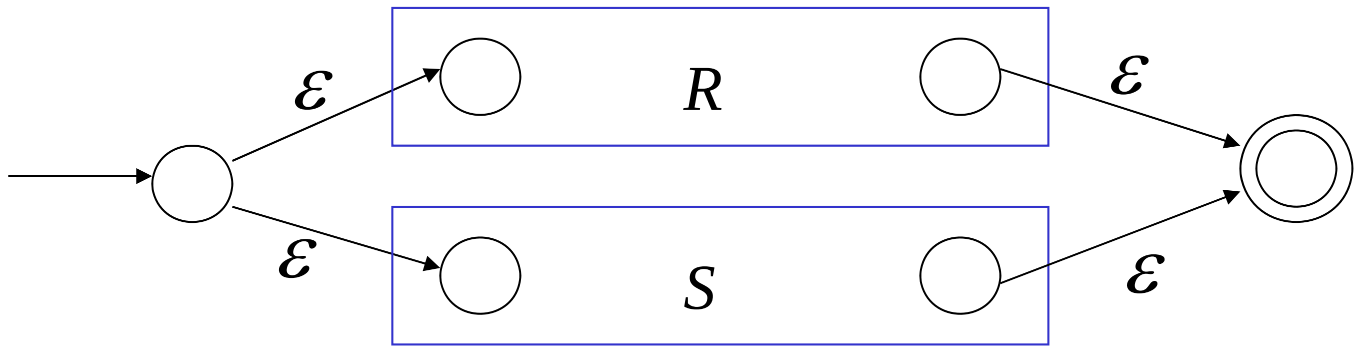
Basis :

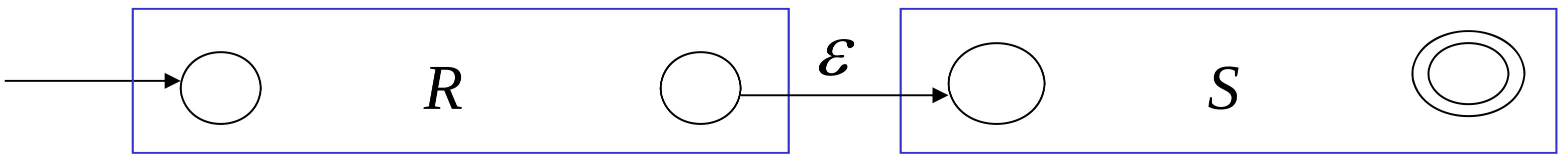


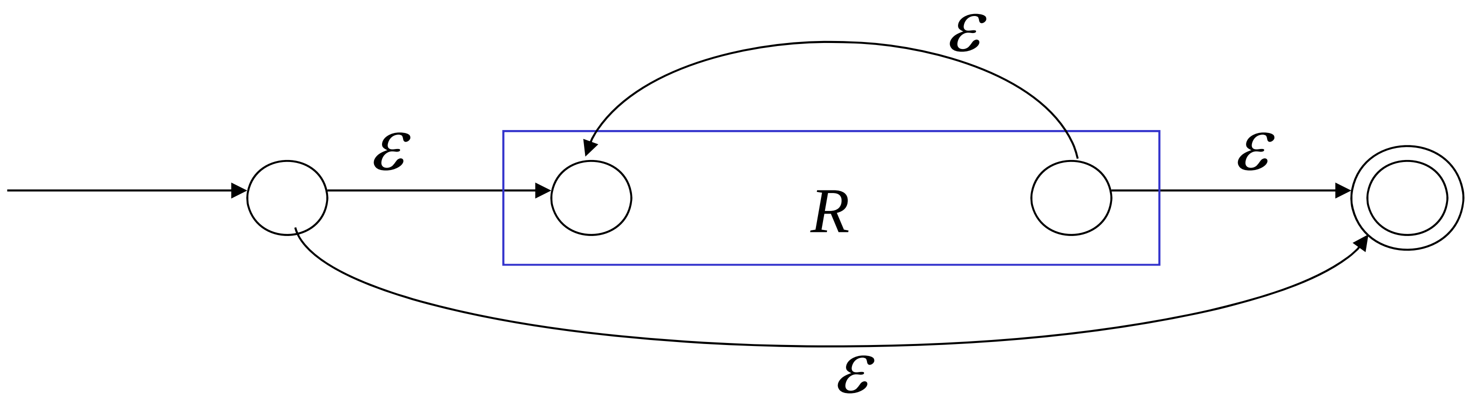


Construct FA from RE

Induction :



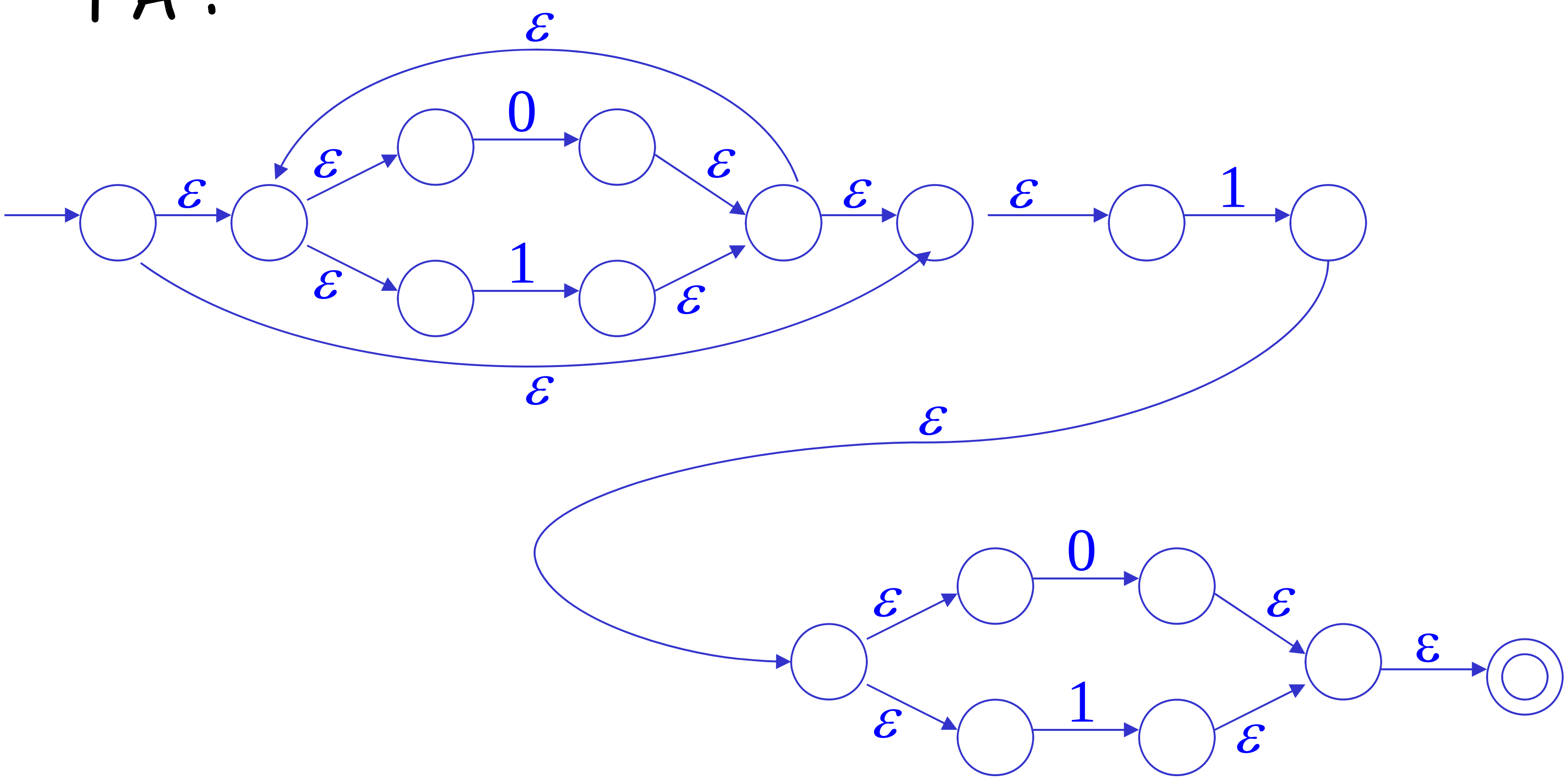




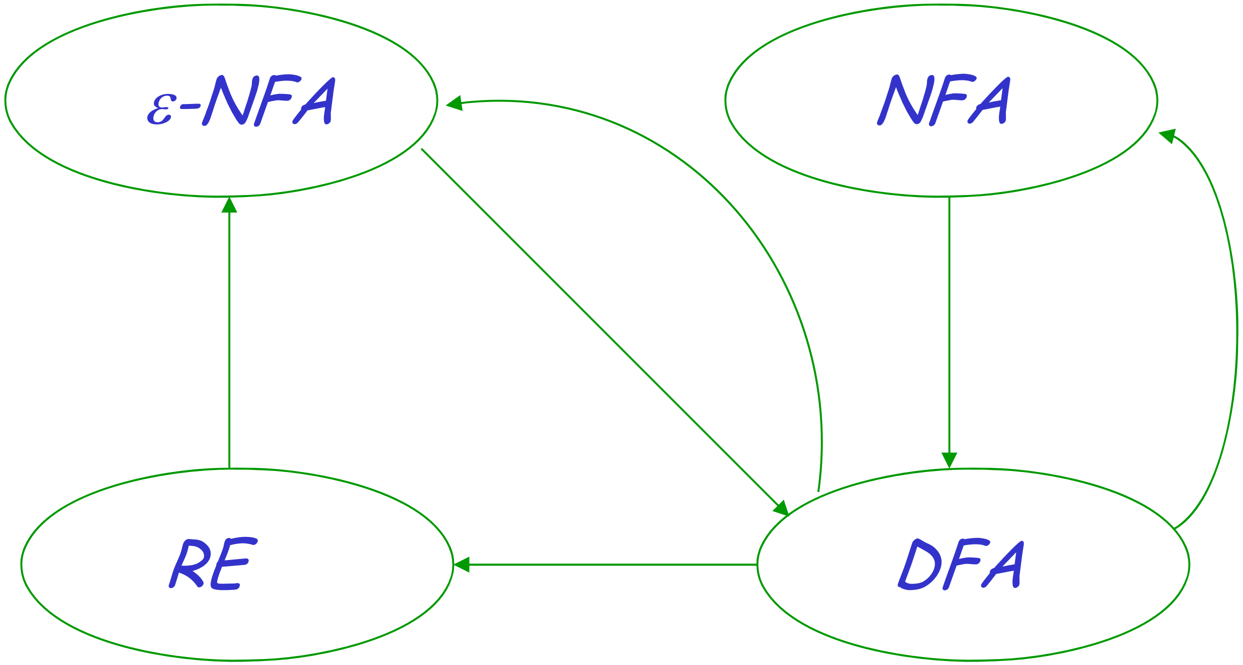
Example 5.8 Construct FA from regular expression

RE : $(0+1)^*1(0+1)$

FA :



FA & RE



What is the equivalence of FAs and REs?

Good good study

day day up !