# Nondeterminism

Deteministic finite automaton

$(Q, \Sigma, \delta, q_0, F)$ $\quad Q = \{q_1, q_2, q_3\}$

$\Sigma = \{a, b\}$

$Q$ set of states

$\Sigma$ alphabet

$\delta : Q \times \Sigma \to Q$ $\quad \delta(q_1, a) = q_3$

$q_0$ initial state

$\cancel{\delta(q_1, a) \neq q_2}$

$F \subseteq Q$ set of final states

# Nondeterministic Finite Automaton (NFA)

DFA vs NFA

1.

$\delta : Q \times \Sigma \to Q$

$\delta_1(q, a) = q_1$

$\delta(q, a) = \{q_1, q_2, \ldots q_n\}$

zero, one, more choices

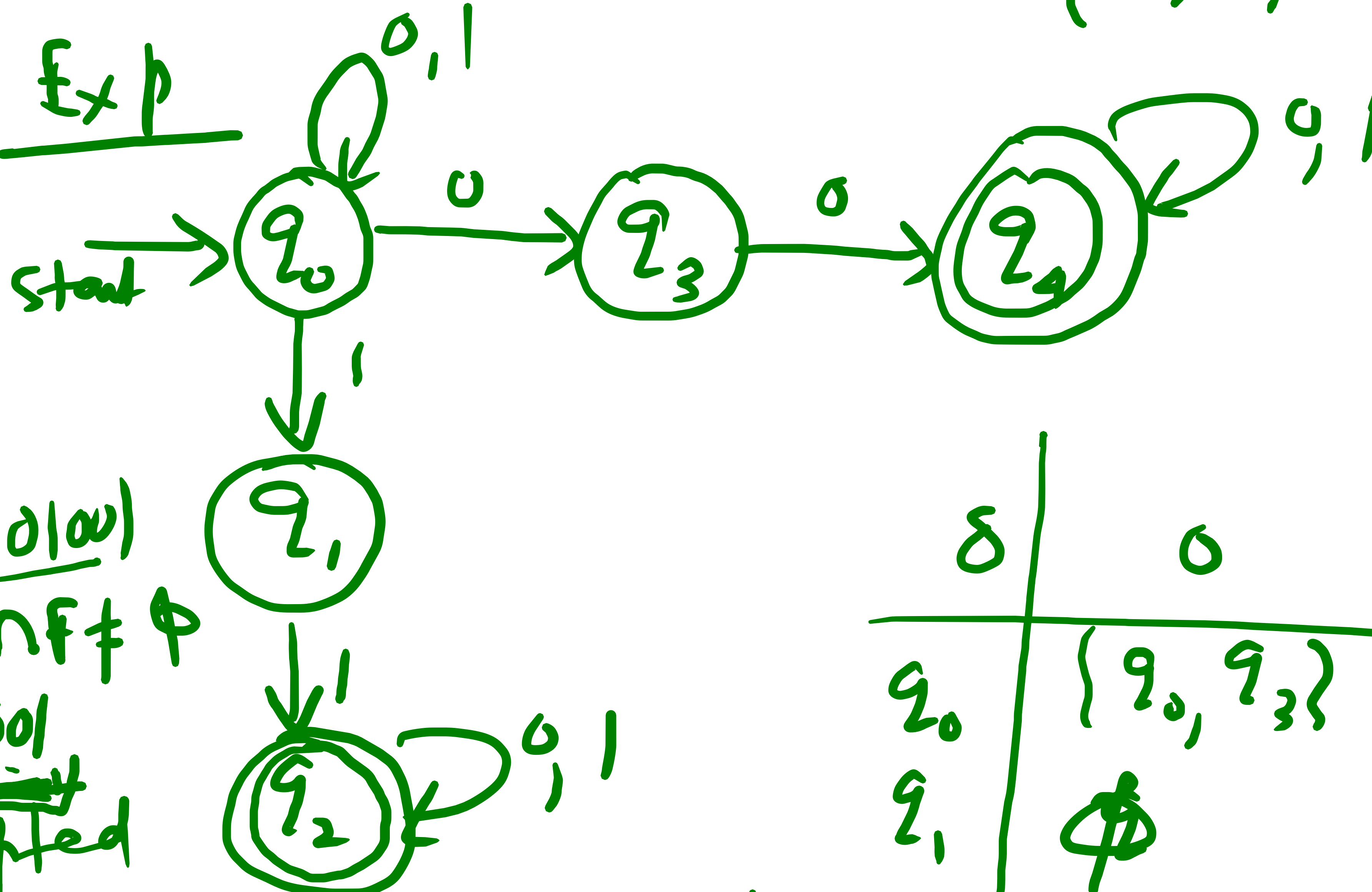
2.

Input symbol is alphabet from $\Sigma$

Input symbol alphabet or $\epsilon$

$\delta(q_0, 0) = \{q_1, q_2, \ldots q_n\}$

Ex




$\delta(q_0, \underline{10100})$

$\cap F \neq \phi$

$w = 01001$

accepted

$\rightarrow q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_0 \xrightarrow{0} q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_0$

$q_0 \xrightarrow{0} q_3$

$q_0 \xrightarrow{1} q_1$

$q_0 \xrightarrow{0} q_3 \xrightarrow{0} q_4 \xrightarrow{1} q_4$

$q_0 \xrightarrow{0} q_3 \xrightarrow{1} \times$

$q_0 \xrightarrow{1} q_1$

| $\delta$ | 0 | 1 |
|---|---|---|
| $q_0$ | $\{q_0, q_3\}$ | $\{q_0, q_1\}$ |
| $q_1$ | $\phi$ | $\{q_2\}$ |
| $q_2$ | $\{q_2\}$ | $\{q_2\}$ |
| $q_3$ | $\{q_4\}$ | $\phi$ |
| $q_4$ | $\{q_4\}$ | $\{q_4\}$ |

# Formal Def$^n$

A NFA is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$
where

$Q$ set of states $\quad Q = \{q_1, q_2, \ldots\}$

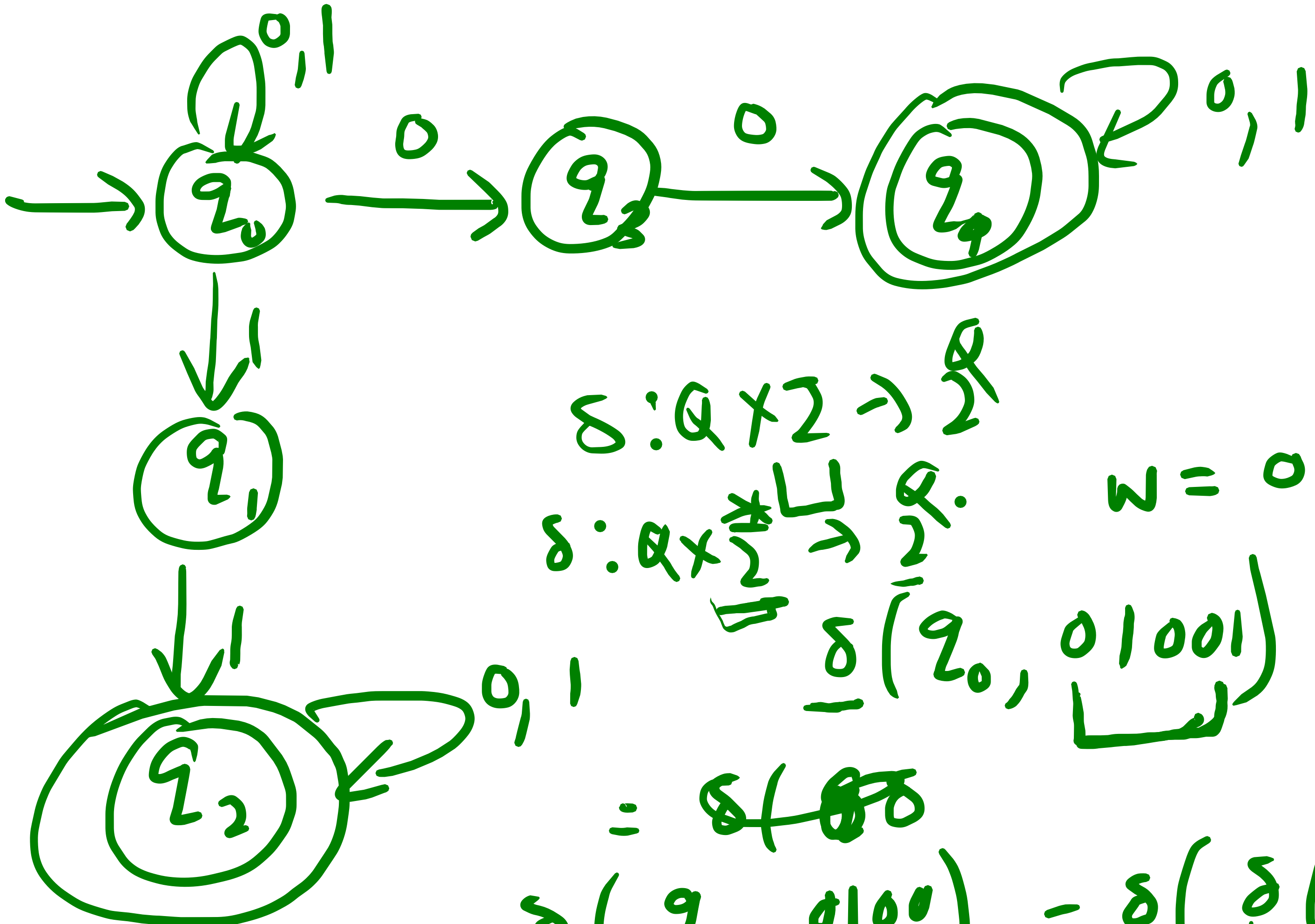
$\Sigma$ finite alphabet $\quad 2^Q = \{\phi, \{q_1\}, \{q_2\}, \{q_1, q_2\}\}$

$\delta : Q \times \Sigma \to 2^Q \quad P(Q)$

$q_0 \in Q$ initial state

$F \subseteq Q$ set of accept states

$\delta(q, a) = \{q_1, q_2, \ldots q_n\}$



$\delta : Q \times \Sigma \to 2^Q$

$\delta : Q \times \Sigma^* \to 2^Q$

$w = 01001$

$\hat{\delta}(q_0, 01001) = \delta(\hat{\delta}(q_0, 0100), 1) = \{q_0, q_1, q_4\}$

$= \cancel{\delta(00}$

$\hat{\delta}(q_0, 0100) = \delta(\hat{\delta}(q_0, 010), 0) = \delta(\{q_0, q_3\}, 0) = \{q_0, q_3, q_4\}$

$\hat{\delta}(q_0, 010) = \delta(\underbrace{\hat{\delta}(q_0, 01)}_{q_0, q_1}, 0) = \delta(\{q_0, q_1\}, 0)$

$= \delta(q_0, 0) \cup \delta(q_1, 0) = \{q_0, q_3\} \cup \emptyset = \{q_0, q_3\}$

$\hat{\delta}(q_0, 01) = \delta(\hat{\delta}(q_0, 0), 1) = \delta(\{q_0, q_3\}, 1)$

$= \delta(q_0, 1) \cup \delta(q_3, 1) = \{q_0, q_1\} \cup \emptyset$

$= \{q_0, q_1\}$

## Extension of $\delta$ to $\hat{\delta}$.

Define $\hat{\delta} : Q \times \Sigma^* \to 2^Q$

1. $\hat{\delta}(q, \epsilon) = \{q\}$

2. $\hat{\delta}(q, wa) = \bigcup_{r \in \hat{\delta}(q,w)} \delta(r, a)$

$= \delta(\underbrace{\hat{\delta}(q,w)}_{}, a)$

$\hat{\delta}(q, w) = \{r_1, r_2 \cdots r_n\}$

3. Define $\delta : 2^Q \times \Sigma^* \to 2^Q$

$\delta(P, w) = \bigcup_{q \in P} \delta(q, w)$ $\quad P \subseteq Q$

$$L(M) = \{ w \mid \delta(q_0, w) \cap F \neq \phi \}$$

$$= \{ w \mid \delta(q_0, w) \text{ contains a state in } F \}$$

$a = a\epsilon$

$$\hat{\delta}(q, a) = \delta(q, a)$$

$$\hat{\delta}(q, \cancel{\epsilon}a)$$

$$= \delta(\hat{\delta}(q, \epsilon), a)$$

$$= \delta$$

$$\underline{L(M)} = \{ w \mid \underbrace{\delta(q_0, \underline{w})} \cap F \neq \phi \}$$

$$= \{ \underline{w} \mid \delta(q_0, w) \text{ contains a state in } \underline{F} \}$$

$q = q\epsilon$

$\delta \quad \hat{\delta}$

$$\underline{\hat{\delta}(q, a)} = \delta(q, a)$$

$$\delta(q_0, \underbrace{01001})$$

$$= \underbrace{\{q_0, q_1, q_4\}} \cap \underbrace{\{q_2, q_4\}}_{F}$$

$$\neq \phi$$

$$\hat{\delta}(q, \cancel{\epsilon} a)$$

$$= \delta(\hat{\delta}(q, \epsilon), a)$$

$$= \underline{\delta(q, a)}$$

Nondeterminism

Deterministic

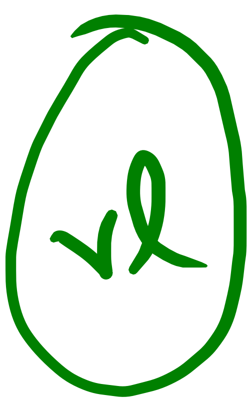
$\delta : Q \times \Sigma \to Q$

$\delta : Q \times \Sigma \to 2^Q$

Every DFA is an NFA

$\Leftarrow$ NFA is a generalization of DFA

regular language




DFA


language accepted NFA

$\underline{Th^m}$ If $L$ is accepted by NFA
Then there exists a DFA that accepts $L$.

$Q = \{q_1, q_2, q_3\}$

$2^Q = \{\phi, \{q_1\}, \{q_2\}, \{q_3\}, \{q_1, q_2\}, \{q_1, q_3\}, \{q_2, q_3\}, \{q_1, q_2, q_3\}\}$

$\underline{M} = (\underline{Q}, \Sigma, \delta, \underline{q_0}, \underline{F})$ be an NFA accepting $L$.

$\delta : Q \times \Sigma \to 2^Q$

3. $q_0' = \underline{\{q_0\}}$

~~Let~~ Define a $\underline{\text{DFA}}$, $M' = (Q', \Sigma, \underline{\delta'}, q_0', \underline{F'})$

1. $Q' = 2^Q$

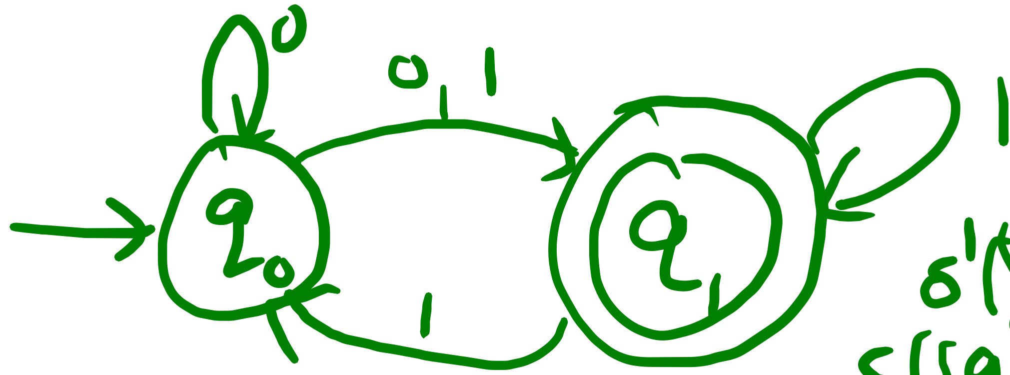
3. $q_0' = \{q_0\}$

2. $\underline{F'} = \{ \underbrace{\{q_1, \ldots q_i\}} \mid \{q_1, q_2 \ldots q_i\}$ contains a final state of $M$

4. $\delta'(\{q_1, q_2, \ldots q_i\}, a) = \{p_1, p_2 \ldots p_j\}$

iff $\delta(\{q_1, q_2, \ldots q_i\}, a) = \{p_1, p_2 \ldots p_j\}$

Exp



$M = (\{q_0, q_1\}, \{0,1\}, \delta, q_0, \{q_1\})$

$\delta(q_0, 0) = \{q_0, q_1\}$, $\delta(q_0, 1) = \{q_1\}$

$\delta(q_1, 0) = \Phi$, $\delta(q_1, 1) = \{q_0, q_1\}$

construct a DFA $M' = (Q', \{0,1\}, \delta', \{q_0\}, F')$

$Q' = 2^Q = \{\Phi, \{q_0\}, \{q_1\}, \{q_0, q_1\}\}$

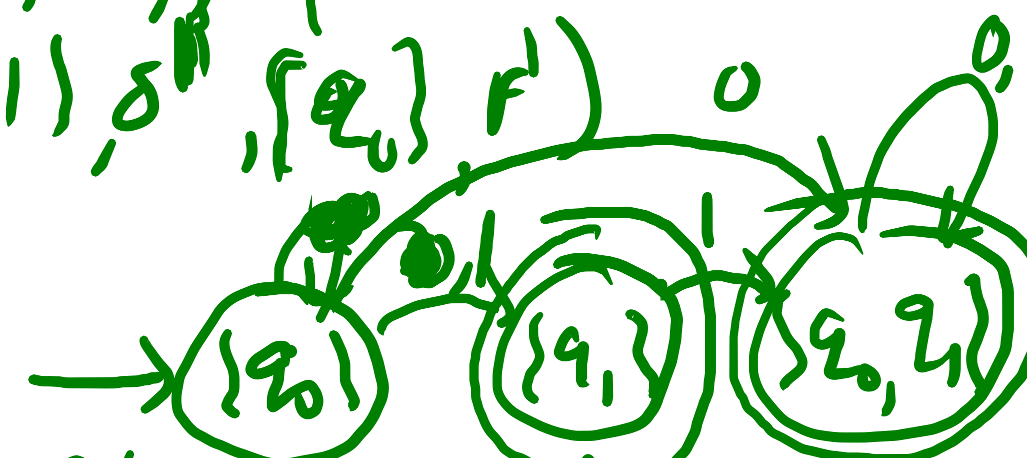
$\delta'(\{q_0\}, 0) = \delta(q_0, 0) = \{q_0, q_1\}$

$\delta'(\{q_0\}, 1) = \delta(q_0, 1) = \{q_1\}$

$\delta'(\{q_1\}, 0) = \delta(q_1, 0) = \Phi$

$\delta'(\{q_1\}, 1) = \delta(q_1, 1) = \{q_0, q_1\}$

$\delta'(\{q_0, q_1\}, 0)$
$= \delta(\{q_0, q_1\}, 0) = \delta(q_0, 0) \cup \delta(q_1, 0)$
$\{q_0, q_1\} \cup \Phi$
$= \{q_0, q_1\}$



$\delta'(\{q_0, q_1\}, 0) =$

$\delta'(\{q_0, q_1\}, 1) =$
$= \delta'$ $\{q_0, q_1\}$

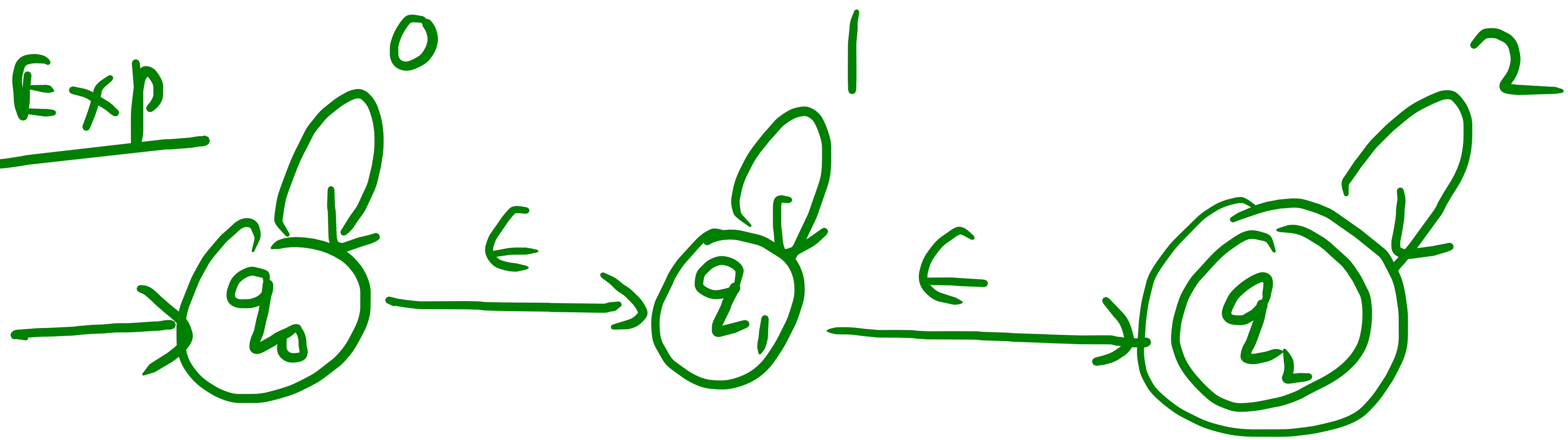
NFA with $\epsilon$ moves

$$(Q, \Sigma, \delta, q_0, F)$$

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

$$\delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q$$

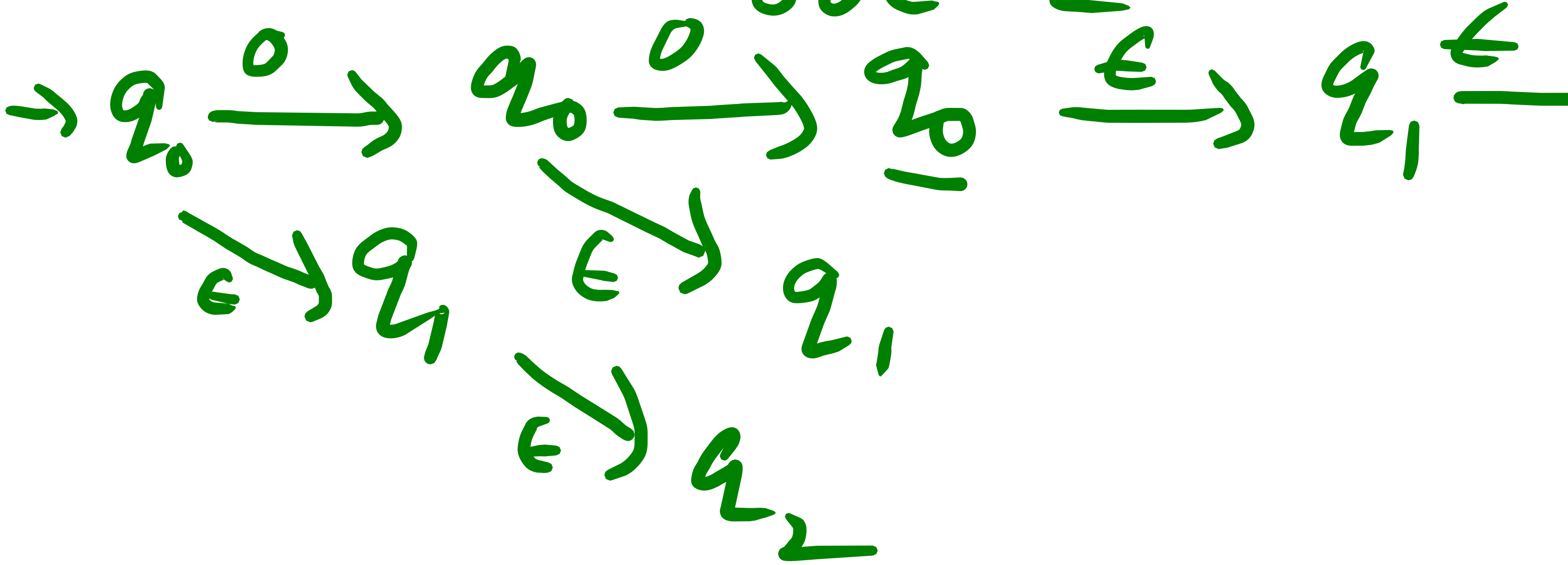
## Exp

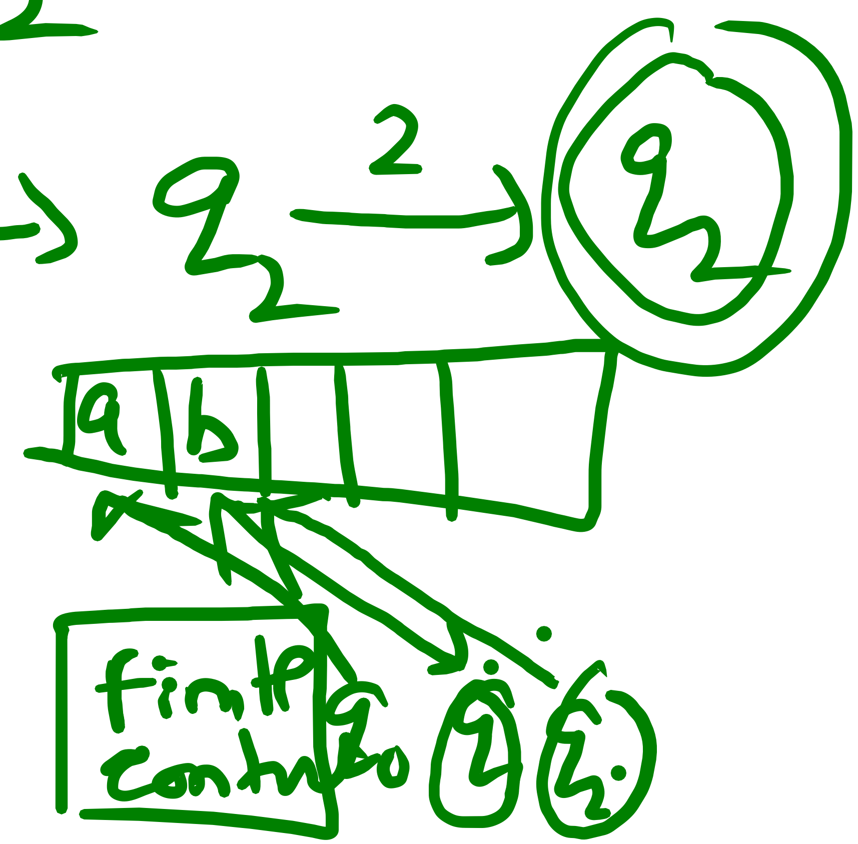



$w = \underline{002}$

$\epsilon 0 \underline{0} 2$

$0 0 \epsilon \quad 2$



NFA

DFA

NFA + ε

regular expression

$(r_1 + r_2) r_1^*$

01, 001, 100

$L(M) = \{ w \mid w \text{ contains a } 1 \}$

set of strings