Automi e Linguaggi Formali

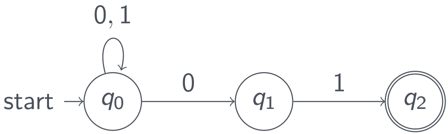
Parte 2 – Automi a Stati Finiti Non Deterministici

UNIVERSITÀ DEGLI STUDI DI PADOVA

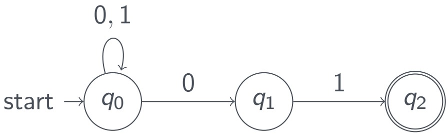
Davide Bresolin
Ultimo aggiornamento: 2 marzo 2021

Sommario

- Cosa fa questo automa?



- Cosa fa questo automa?



- È un esempio di automa a stati finiti non deterministico:
  - può trovarsi contemporaneamente in più stati diversi
  - le transizioni non sono necessariamente complete:
    - da q_1 si esce solo leggendo 1
    - q_2 non ha transizioni uscenti

  in questi casi il percorso si blocca, ma può proseguire lungo gli altri percorsi

Definizione formale di NFA

Un Automa a Stati Finiti Non Deterministico (NFA) è una quintupla

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

- Q è un insieme finito di stati
- Σ è un alfabeto finito (= simboli in input)
- δ è una funzione di transizione che prende in input (q, a) e restituisce un sottoinsieme di Q
- $q_0 \in Q$ è lo stato iniziale
- $F \subseteq Q$ è un insieme di stati finali

Tabella delle transizioni per l'esempio

L'NFA che riconosce le parole che terminano con 01 è

$$A = (Q, \{0, 1\}, \delta, q_0, \{q_2\})$$

dove δ è la funzione di transizione

	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\emptyset$	$\{q_2\}$
$*q_2$	$\emptyset$	$\emptyset$

Computazione di un NFA

- Data una parola $w = w_1 w_2 \dots w_n$, una computazione di un NFA A con input w è una sequenza di stati $r_0 r_1 \dots r_n$ che rispetta due condizioni:
  1. $r_0 = q_0$ (inizia dallo stato iniziale)
  2. $\delta(r_i, w_{i+1}) = r_{i+1}$ per ogni $i = 0, \dots, n-1$ (rispetta la funzione di transizione)
- Diciamo che una computazione accetta la parola w se:
  3. $r_n \in F$ (la computazione termina in uno stato finale)
- A causa del nondeterminismo, ci può essere più di una computazione per ogni parola!

Linguaggio accettato da un NFA

- Un NFA A accetta la parola w se esiste una computazione che accetta w
- Un NFA A rifiuta la parola w se tutte le computazioni la rifiutano
- Formalmente, il linguaggio accettato da A è

$$L(A) = \{w \in \Sigma^* \mid A \text{ accetta } w\}$$

Esercizi

Definire degli automi a stati finiti non deterministici che accettino i seguenti linguaggi:

- L'insieme delle parole sull'alfabeto $\{0, 1, \ldots, 9\}$ tali che la cifra finale sia comparsa in precedenza
- L'insieme delle parole sull'alfabeto $\{0, 1, \ldots, 9\}$ tali che la cifra finale *non* sia comparsa in precedenza
- L'insieme delle parole di 0 e 1 tali che esistono due 0 separati da un numero di posizioni multiplo di 4 (0 è un multiplo di 4)

Esercizio

Università degli Studi di Padova

Consideriamo l'alfabeto $\Sigma = \{a, b, c, d\}$ e costruiamo un automa non deterministico che riconosce il linguaggio di tutte le parole tali che uno dei simboli dell'alfabeto non compare mai:

- tutte le parole che non contengono a
- $+$ tutte le parole che non contengono b
- $+$ tutte le parole che non contengono c
- $+$ tutte le parole che non contengono d

Dopo aver costruito l'NFA, trasform

Equivalenza di DFA e NFA

- Sorprendentemente, NFA e DFA sono in grado di riconoscere gli stessi linugaggi
- Per ogni NFA N c'è un DFA D tale che $L(D) = L(N)$, e viceversa
- L'equivalenza di dimostra mediante una costruzione a sottoinsiemi:

Equivalenza di DFA e NFA

UNIVERSITÀ DEGLI STUDI DI PADOVA

- Sorprendentemente, NFA e DFA sono in grado di riconoscere gli stessi linugaggi
- Per ogni NFA N c'è un DFA D tale che $L(D) = L(N)$, e viceversa
- L'equivalenza di dimostra mediante una costruzione a sottoinsiemi:
  Dato un NFA
  $$N = (Q_N, \Sigma, q_0, \delta_N, F_N)$$
  costruiremo un DFA
  $$D = (Q_D, \Sigma, \{q_0\}, \delta_D, F_D)$$
  tale che
  $$L(D) = L(N)$$

La costruzione a sottoinsiemi

- $Q_D = \{S : S \subseteq Q_N\}$
  Ogni stato del DFA corrisponde ad un insieme di stati dell'NFA
- $F_D = \{S \subseteq Q_N : S \cap F_N \neq \emptyset\}$
  Uno stato del DFA è finale se c'è almeno uno stato finale corrispondente nell'NFA
- Per ogni $S \subseteq Q_N$ e per ogni $a \in \Sigma$

$$\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$$

La funzione di transizione "percorre tutte le possibili strade"

La costruzione a sottoinsiemi

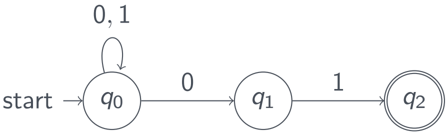
- $Q_D = \{S : S \subseteq Q_N\}$
  Ogni stato del DFA corrisponde ad un insieme di stati dell'NFA
- $F_D = \{S \subseteq Q_N : S \cap F_N \neq \emptyset\}$
  Uno stato del DFA è finale se c'è almeno uno stato finale corrispondente nell'NFA
- Per ogni $S \subseteq Q_N$ e per ogni $a \in \Sigma$

$$\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$$

  La funzione di transizione "percorre tutte le possibili strade"

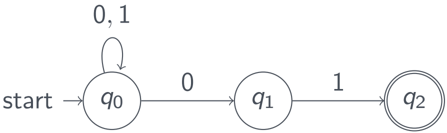
Nota: $|Q_D| = 2^{|Q_N|}$, anche se spesso la maggior parte degli stati in Q_D sono "inutili", cioè non raggiungibili dallo stato iniziale.

Esempio di costruzione a sottoinsiemi



Costruiamo δ_D per l'NFA qui sopra:

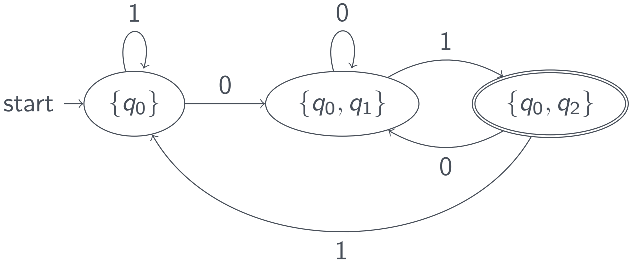
Esempio di costruzione a sottoinsiemi



Costruiamo δ_D per l'NFA qui sopra:

	0	1
$\emptyset$	$\emptyset$	$\emptyset$
$\rightarrow \{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_1\}$	$\emptyset$	$\{q_2\}$
$*\{q_2\}$	$\emptyset$	$\emptyset$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$*\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_0\}$
$*\{q_1, q_2\}$	$\emptyset$	$\{q_2\}$
$*\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$

La tabella di transizione per D ci permette di ottenere il diagramma di transizione



Per semplificare il disegno, ho omesso gli stati non raggiungibili

1. Determinare il DFA equivalente all'NFA con la seguente tabella di transizione:

	0	1
$\rightarrow q_0$	$\{q_0\}$	$\{q_0, q_1\}$
q_1	$\{q_1\}$	$\{q_0, q_2\}$
$*q_2$	$\{q_1, q_2\}$	$\{q_0, q_1, q_2\}$

2. Qual è il linguaggio accettato dall'automa?

Teorema di equivalenza tra DFA e NFA

Theorem

Un linguaggio L è accettato da un DFA se e solo se è accettato da un NFA.

Dimostrazione:

- La parte "se" è data dalla costruzione per sottoinsiemi
- La parte "solo se" si dimostra osservando che ogni DFA può essere trasformato in un NFA modificando δ_D in δ_N con la seguente regola:

  Se $\delta_D(q, a) = p$ allora $\delta_N(q, a) = \{p\}$

Esercizio 1

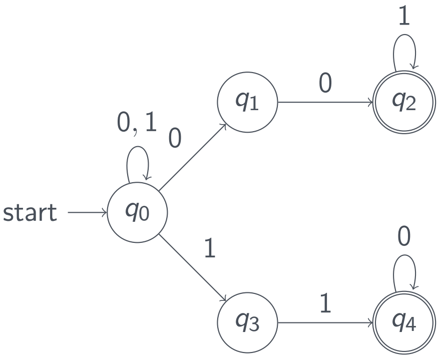
1. Determinare il DFA equivalente all'NFA con la seguente tabella di transizione:

	0	1
$\rightarrow q_0$	$\{q_0\}$	$\{q_0, q_1\}$
q_1	$\{q_1\}$	$\{q_0, q_2\}$
$*q_2$	$\{q_1, q_2\}$	$\{q_0, q_1, q_2\}$

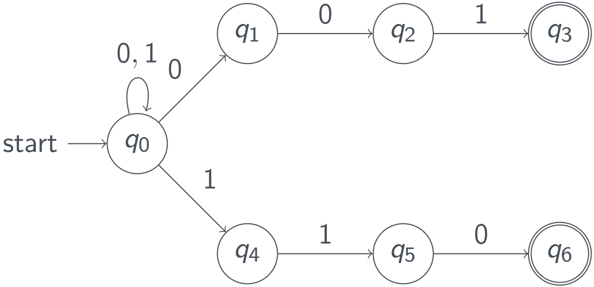
2. Qual è il linguaggio accettato dall'automa?

Esercizio 2

Trasformare il seguente NFA in DFA



Dato il seguente NFA



1. determinare il linguaggio riconosciuto dall'automa
2. costruire un DFA equivalente

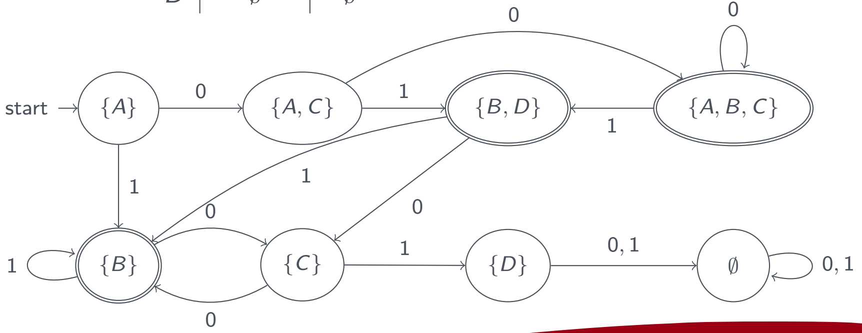
Esercizio

Convertire il seguente NFA in DFA:

	0	1
$\rightarrow A$	$\{A, C\}$	$\{B\}$
$*B$	$\{C\}$	$\{B\}$
C	$\{B\}$	$\{D\}$
D	$\emptyset$	$\emptyset$

Convertire il seguente NFA in DFA:

	0	1
$\rightarrow A$	$\{A, C\}$	$\{B\}$
$*B$	$\{C\}$	$\{B\}$
C	$\{B\}$	$\{D\}$
D	$\emptyset$	$\emptyset$



Sommario

NFA con epsilon-transizioni

Esercizio: costruiamo un NFA che accetta numeri decimali:

1. Un segno $+$ o $-$, opzionale
2. Una stringa di cifre decimali $\{0, \ldots, 9\}$
3. un punto decimale .
4. un'altra stringa di cifre decimali

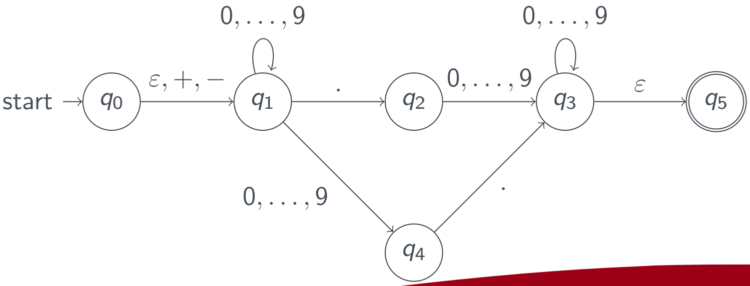
Una delle stringhe (2) e (4) può essere vuota, ma non entrambe

NFA con epsilon-transizioni

Esercizio: costruiamo un NFA che accetta numeri decimali:

1. Un segno $+$ o $-$, opzionale
2. Una stringa di cifre decimali $\{0, \ldots, 9\}$
3. un punto decimale .
4. un'altra stringa di cifre decimali

Una delle stringhe (2) e (4) può essere vuota, ma non entrambe



ε-NFA: definizione

Un Automa a Stati Finiti Non Deterministico con ε-transizioni (ε-NFA) è una quintupla

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

dove:

- Q, Σ, q_0, F sono definiti come al solito
- δ è una funzione di transizione che prende in input:
  - uno stato in Q
  - un simbolo nell'alfabeto $\Sigma \cup \{\varepsilon\}$

  e restituisce un sottoinsieme di Q

Esempio di ε-NFA

L'automa che riconosce le cifre decimali è definito come

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

dove δ è definita dalla tabella di transizione

Esempio di ε-NFA

L'automa che riconosce le cifre decimali è definito come

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

dove δ è definita dalla tabella di transizione

	ε	$+, -$	$.$	$0, \ldots, 9$
$\rightarrow q_0$	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
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	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
$*q_5$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Epsilon chiusura: definizione

L'eliminazione delle ε-transizioni procede per ε-chiusura degli stati:

- tutti gli stati raggiungibili da q con una sequenza $\varepsilon\varepsilon\ldots\varepsilon$

La definizione di $\mathrm{ECLOSE}(q)$ è per induzione:

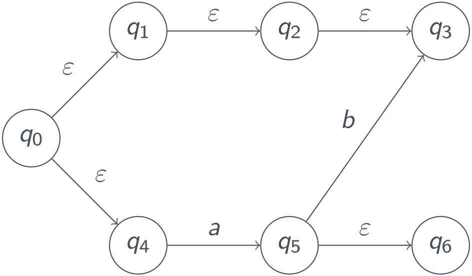
Caso base:

$$q \in \mathrm{ECLOSE}(q)$$

Caso induttivo:

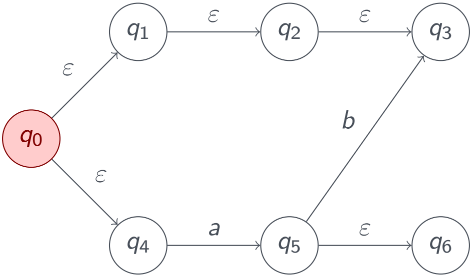
$$\text{se } p \in \mathrm{ECLOSE}(q) \text{ e } r \in \delta(p, \varepsilon) \text{ allora } r \in \mathrm{ECLOSE}(q)$$

Epsilon chiusura: esempio

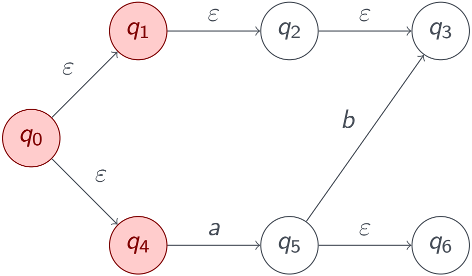


$$\mathrm{ECLOSE}(q_0) = \{$$

Epsilon chiusura: esempio

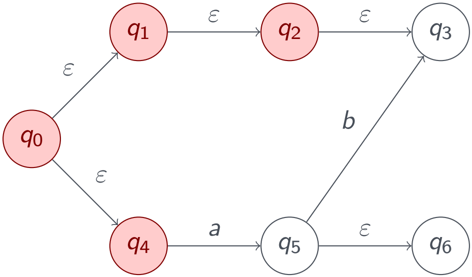


$$\mathrm{ECLOSE}(q_0) = \{q_0$$



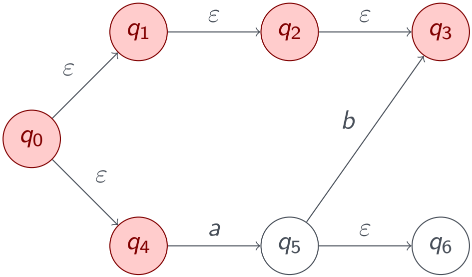
$$\mathrm{ECLOSE}(q_0) = \{q_0, q_1, q_4$$

Epsilon chiusura: esempio



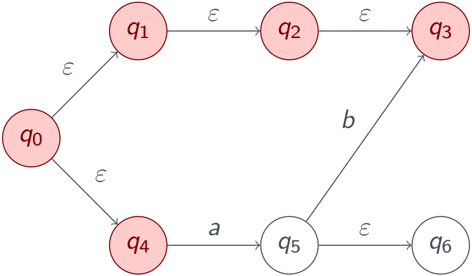
$$\mathrm{ECLOSE}(q_0) = \{q_0, q_1, q_4, q_2$$

Epsilon chiusura: esempio



$$\text{ECLOSE}(q_0) = \{q_0, q_1, q_4, q_2, q_3$$

Epsilon chiusura: esempio



$$\mathrm{ECLOSE}(q_0) = \{q_0, q_1, q_4, q_2, q_3\}$$

Equivalenza di DFA e ε-NFA

- Anche in questo caso abbiamo definito una classe di automi che è equivalente ai DFA
- Per ogni ε-NFA E c'è un DFA D tale che $L(E) = L(D)$, e viceversa
- Lo si dimostra modificando la costruzione a sottoinsiemi:

Equivalenza di DFA e ε-NFA

- Anche in questo caso abbiamo definito una classe di automi che è equivalente ai DFA
- Per ogni ε-NFA E c'è un DFA D tale che $L(E) = L(D)$, e viceversa
- Lo si dimostra modificando la costruzione a sottoinsiemi: Dato un ε-NFA

$$E = (Q_E, \Sigma, q_0, \delta_E, F_E)$$

costruiremo un DFA

$$D = (Q_D, \Sigma, S_0, \delta_D, F_D)$$

tale che

$$L(D) = L(E)$$

La costruzione a sottoinsiemi modificata

- $Q_D = \{S \subseteq Q_E : S = \mathrm{ECLOSE}(S)\}$
  Ogni stato è un insieme di stati chiuso per ε-transizioni
- $S_0 = \mathrm{ECLOSE}(q_0)$
  Lo stato iniziale è la ε-chiusura dello stato iniziale di E
- $F_D = \{S \in Q_D : S \cap F_E \neq \emptyset\}$
  Uno stato del DFA è finale se c'è almeno uno stato finale di E
- Per ogni $S \in Q_D$ e per ogni $a \in \Sigma$:

$$\delta_D(S, a) = \mathrm{ECLOSE}\Big(\bigcup_{p \in S} \delta_E(p, a)\Big)$$

  La funzione di transizione "percorre tutte le possibili strade" (comprese quelle con ε-transizioni)

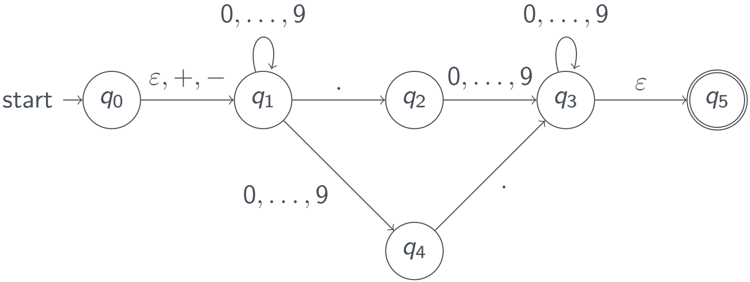
- $Q_D = \{S \subseteq Q_E : S = \mathrm{ECLOSE}(S)\}$
  Ogni stato è un insieme di stati chiuso per ε-transizioni
- $S_0 = \mathrm{ECLOSE}(q_0)$
  Lo stato iniziale è la ε-chiusura dello stato iniziale di E
- $F_D = \{S \in Q_D : S \cap F_E \neq \emptyset\}$
  Uno stato del DFA è finale se c'è almeno uno stato finale di E
- Per ogni $S \in Q_D$ e per ogni $a \in \Sigma$:

$$\delta_D(S, a) = \mathrm{ECLOSE}\left(\bigcup_{p \in S} \delta_E(p, a)\right)$$

  La funzione di transizione "percorre tutte le possibili strade" (comprese quelle con ε-transizioni)

Nota: anche in questo caso $|Q_D| = 2^{|Q_E|}$

Costruiamo un DFA D equivalente all'ε-NFA E che riconosce i numeri decimali:



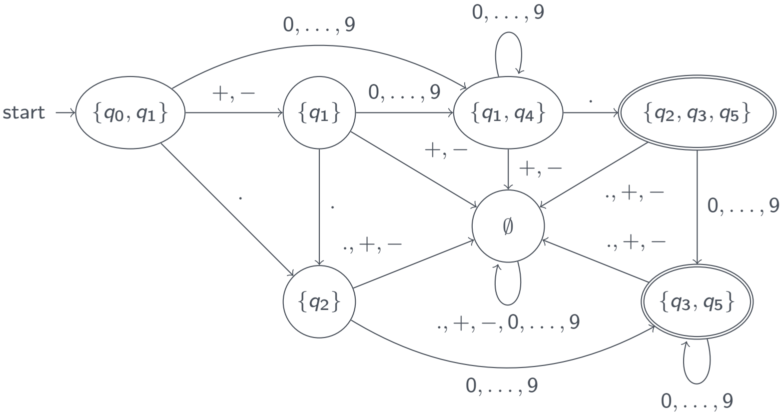
Esempio di costruzione a sottoinsiemi (2)

- Come prima cosa costruiamo la ε-chiusura di ogni stato:

$$
\begin{array}{ll}
\mathrm{ECLOSE}(q_0) = \{q_0, q_1\} & \mathrm{ECLOSE}(q_1) = \{q_1\} \\
\mathrm{ECLOSE}(q_2) = \{q_2\} & \mathrm{ECLOSE}(q_3) = \{q_3, q_5\} \\
\mathrm{ECLOSE}(q_4) = \{q_4\} & \mathrm{ECLOSE}(q_5) = \{q_5\}
\end{array}
$$

- Lo stato iniziale di D è $\{q_0, q_1\}$

- Applicando le regole otteniamo il diagramma di transizione:



Teorema di equivalenza tra DFA e NFA

Theorem

Un linguaggio L è accettato da un DFA se e solo se è accettato da un ε-NFA.

Dimostrazione:

- La parte "se" è data dalla costruzione per sottoinsiemi modificata
- La parte "solo se" si dimostra osservando che ogni DFA può essere trasformato in un ε-NFA modificando δ_D in δ_E con la seguente regola:

  Se $\delta_D(q, a) = p$ allora $\delta_E(q, a) = \{p\}$

Esercizio

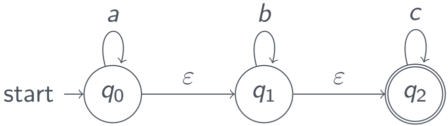
1. Costruiamo un ε-NFA che riconosce le parole costituite da
   - zero o più a
   - seguite da zero o più b
   - seguite da zero o più c
2. Calcolare ECLOSE di ogni stato dell'automa
3. Convertire l'ε-NFA in DFA

1. Costruiamo un ε-NFA che riconosce le parole costituite da zero o più *a*, seguite da zero o più *b*, seguite da zero o più *c*

2. Calcolare la ε-chiusura di ogni stato

3. Convertire l'ε-NFA in DFA

1. Costruiamo un ε-NFA che riconosce le parole costituite da zero o più a, seguite da zero o più b, seguite da zero o più c

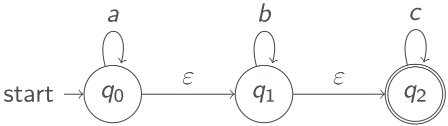


2. Calcolare la ε-chiusura di ogni stato

3. Convertire l'ε-NFA in DFA

Esercizio

1. Costruiamo un ε-NFA che riconosce le parole costituite da zero o più a, seguite da zero o più b, seguite da zero o più c



2. Calcolare la ε-chiusura di ogni stato

$$\mathrm{ECLOSE}(q_0) = \{q_0, q_1, q_2\}$$
$$\mathrm{ECLOSE}(q_1) = \{q_1, q_2\}$$
$$\mathrm{ECLOSE}(q_2) = \{q_2\}$$

3. Convertire l'ε-NFA in DFA

Esercizio – continua

1. Costruiamo un ε-NFA che riconosce le parole costituite da zero o più a, seguite da zero o più b, seguite da zero o più c
2. Calcolare la ε-chiusura di ogni stato
3. Convertire l'ε-NFA in DFA