## Logica e Modelli Computazionali

# Macchine di Turing

**Marco Console** 

Ingegneria Informatica e Automatica (Sapienza, Università di Roma)

#### Oltre i Linguaggi Non-Contestuali

- Fino ad ora abbiamo visto due diversi modelli computazionali
  - Automi a stati finiti (D, ND,  $\epsilon$ ). L'unica memoria che posseggono è lo stato interno
  - Automi a Pila. Posseggono memoria illimitata ma gestita come una Pila (LIFO Queue)
- Abbiamo visto che entrambi i modelli hanno delle limitazioni molto stringenti
  - Possiamo provare un Pumping Lemma (distinto) per entrambi i modelli
  - II linguaggio  $\mathcal{L}_1 = \{a^m b^m \mid m \geq 1\}$  non è regolare (non possiamo riconoscerlo  $\epsilon$ -ASFND)
  - II linguaggio  $\mathcal{L}_2 = \{a^m b^m c^m | m \ge 1\}$  non è Non-Contestuale (non sappiamo riconoscerlo!)
- Possiamo aumentare il potere computazionale delle nostre macchine rendendo più libero l'accesso alla memoria durante la computazione
  - Il linguaggio $\mathcal{L}_2$  può essere riconosciuto semplicemente "contando" le occorrenze dei tre simboli
  - Per farlo dobbiamo però tenere traccia del numero di simboli incontrati senza dimenticarlo

### Macchine di Turing – Intuizione

- Per costruire un modello matematico con accesso alla memoria privo di restrizioni ci viene in aiuto un modello computazionale inventato negli anni 1920 dal matematico britannico Alan Turing
- Le definizioni della Macchina di Turing sono molte (e tutte equivalenti a meno di parametri) ma si basano tutte sull'idea di fornire ad un Automa a Stati Finiti le seguenti caratteristiche aggiuntive
  - La macchina ha un insieme finito di stati interni tra cui si muove (come gli ASFD)
  - La macchina ha accesso a una memoria illimitata (come gli automi a pila)
  - L'input non è consumato durante la computazione ma risiede nella memoria interna
  - La memoria è un nastro illimitato (stringa infinita) le cui celle (caratteri) possono essere letti e\o scritti
  - La macchina può terminare la sua computazione in qualunque momento senza consumare tutto l'input
- In termini di linguaggi riconosciuti, tale modello computazionale è il più potente che conosciamo
  - In termini di linguaggi riconosciuti, congetturiamo che non sia possibile trovare di meglio ©
  - "It is amazing how little we need to have everything." (Christos H. Papadimitriou Computational Complexity)

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

#### **NASTRO**

a a b b c c u u ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
    - **Torna** all'ultima occorrenza del carattere ↓

**NASTRO** 

a a b b c c u u ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
    - **Torna** all'ultima occorrenza del carattere ↓

**NASTRO** 

 $\downarrow$  a b b c c  $\sqcup$   $\sqcup$   $\sqcup$ 

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

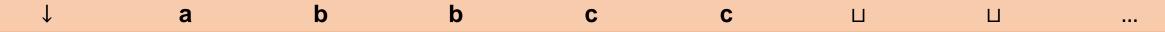
NASTI

 $\downarrow$  a b b c c  $\sqcup$   $\sqcup$   $\ldots$ 

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓





Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓





Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

#### **NASTRO**

Algoritmo per riconoscere 
$$L = \{a^m b^m c^m | m \ge 0\}$$

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere 
$$L = \{a^m b^m c^m | m \ge 0\}$$

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #

Torna all'ultima occorrenza del carattere ↓

**NASTRO** 

↓ a # b # c ⊔ ⊔ ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

NASTI

↓ a # b # c □ □ □ ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓





Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓





Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

#

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente

#

2. Altrimenti cambia tale occorrenza di c con #

#

**4.** Torna all'ultima occorrenza del carattere ↓

#### **NASTRO**



#

Ш

Ш

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

#

#

Ш

Ш

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente

#

2. Altrimenti cambia tale occorrenza di c con #

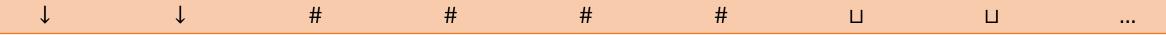
#

**4.** Torna all'ultima occorrenza del carattere ↓

NAST

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

#### **NASTRO**







#

#







•••

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

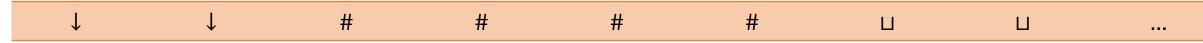


Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

**NASTRO** 

Accetta la stringa corrente



Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

#### **NASTRO**

a a c c b b u u ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
    - **Torna** all'ultima occorrenza del carattere ↓

**NASTRO** 

a a c c b b u u ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
    - Torna all'ultima occorrenza del carattere ↓

**NASTRO** 

 $\downarrow$  a c c b b  $\sqcup$   $\sqcup$   $\sqcup$ 

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

NASTI

 $\downarrow$  a c c b b  $\sqcup$   $\sqcup$  ...

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



 $\downarrow$  a c c b b  $\sqcup$   $\sqcup$   $\ldots$ 

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

**NASTRO** 

Rifiuta la stringa corrente

 ↓
 a
 c
 c
 b
 b
 ⊔
 ⊔
 u
 ...

### **Macchina di Turing – Definizione Formale**

**Definizione**. Una macchina di Turing M è una 7-upla  $M = \langle \Sigma, \Gamma, Q, \delta, q_0, q_{ves}, q_{no} \rangle$  tale che:

- Σ è un insieme finito di simboli detto, insieme dei simboli di input che non include il simbolo blank ⊔
- Γ con Σ ⊆ Γ è un insieme finito di simboli detto, insieme dei simboli del nastro, che include il simbolo blank ⊔
- Q è un insieme finito e non vuoto di stati tale che
  - $q_0 \in Q$  è lo stato iniziale
  - $q_{yes} \in Q$  è lo stato accettante
  - $q_{no} \in Q \text{ con } q_{ves} \neq q_{no} \text{ è lo stato rifiutante}$
  - $Q' = Q \setminus \{q_{yes}, q_{no}\}$  (Q' non contiene  $q_{yes}$  e  $q_{no}$ )
- δ è la funzione di transizione; ovvero, una funzione totale definita come segue  $\delta: Q' \times \Sigma \to Q \times \Sigma \times \{\leftarrow, \rightarrow, -\}$

### Macchina di Turing – Definizione Informale

- Intuizione 1. La definizione che abbiamo dato cerca di catturare il comportamento della macchina che abbiamo descritto fino ad ora
  - La macchina ha diversi stati interni e inizia la computazione dallo stato iniziale
  - Lo stato accettante termina la computazione della macchina e accetta
     l'input
  - Lo stato rifiutante termina la computazione della macchina e rifiuta l'input
  - La macchina può manipolare i simboli della memoria interna e quelli dell'input

#### Macchina di Turing – Definizione Informale

- Intuizione 2. Il prossimo passo di una Macchina di Turing è determinato dallo stato interno corrente, dal contenuto del nastro e dalla posizione della testina nel nastro
  - Da queste informazioni possiamo definire la sequenza eseguiti dalla macchina (computazione)
  - La funzione di transizione è definita per ogni coppia ogni coppia  $(q, \sigma)$  ∈  $Q \times \Sigma$  che rappresenta lo stato interno della macchina e il simbolo correntemente letto dalla testina sul nastro e restituisce una terna  $(q', \sigma', t)$  ∈  $Q \times \Sigma \times \{\leftarrow, \rightarrow, -\}$  dove
    - q' è il prossimo stato interno della macchina (chiaramente non è necessario che  $q' \neq q$ )
    - $\sigma'$  è il simbolo che sovrascrive quello corrente (chiaramente non è necessario che  $\sigma' \neq \sigma$ )
    - t è lo spostamento della testina lungo il nastro
      - ←, spostamento a sinistra (torna indietro di una cella)
      - →, spostamento a destra (vai avanti di una cella)
      - –, nessuno spostamento (rimani alla cella corrente)

- Definiamo la macchina di Turing  $M = < \Sigma, \Gamma, Q, \delta, q_0, q_{yes}, q_{no} >$  come segue
  - $-\Sigma = \{a, b\}$  Simboli di input
  - $\Gamma$  =  $\Sigma$  ∪ {#,⊔} Simboli del nastro
  - $Q = \{q_0, q_1, q_{ves}, q_{no}\}$  Insieme degli stati

$oldsymbol{q}$	σ	$oldsymbol{\delta(q,\sigma)}$
$q_0$	а	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_1, b, \rightarrow)$
$q_0$	#	$(q_{no}, \#, -)$
$q_0$	Ц	$(q_{no},\sqcup,-)$
$q_1$	а	$(q_{no},a,-)$
$q_1$	b	$(q_1, b, \rightarrow)$
$q_1$	#	$(q_{no}, \#, -)$
$q_1$	Ц	$(q_{yes},\sqcup,-)$

## Macchina di Turing – Configurazioni

- Sia  $M = \langle \Sigma, \Gamma, Q, \delta, q_0, q_{ves}, q_{no} \rangle$  una macchina di Turing
- **Definizione**. Una **configurazione** di *M* è una 3-upla  $C = (\sigma, q, \tau)$  tale che:
  - σ è una **stringa sull'alfabeto**  $\Gamma$  (nastro a sinistra della testina)
  - $-\tau = a\tau'$  è una stringa sull'alfabeto  $\Gamma$  (nastro a destra della testina e a è il simbolo corrente)
  - $-q \in Q$  è uno **stato** di M (stato corrente della configurazione)
- Definizione. Una configurazione  $C = (\sigma, q, \tau)$  di M è detta:
  - Accettante (finale) se  $q = q_{ves}$
  - Rifiutante (finale) se  $q = q_{no}$
  - Iniziale se se  $q=q_0$ ,  $\sigma=\epsilon$  (stringa vuota) e  $\tau\in\Sigma^*$  ( $\tau$  è una stringa dell'alfabeto dell'input)

## Macchina di Turing – Configurazioni

- Esempio. Sia  $M = < \Sigma, \Gamma, Q, \delta, q_0, q_{yes}, q_{no} > \text{con } \Sigma = \{a, b\}, \Gamma = \Sigma \cup \{\#, \ \sqcup\}, Q = \{q_0, q_1, q_{yes}, q_{no}\}$ 
  - La macchina definita in precedenza
  - $(\epsilon, q_0, aa)$  è una configurazione **Iniziale** di M
  - $(abab, q_{ves}, abb \sqcup)$  è una configurazione **Accettante** di M
  - $(ab, q_{no}, ab \sqcup b)$  è una configurazione Rifiutante di M
  - $-(\epsilon, q_1, aa)$  è una configurazione di M (non iniziale, non finale)
  - $(\epsilon, q_0, aa\#)$  è una configurazione di M (non è iniziale perché  $\# \notin \Sigma$ )

#### Macchina di Turing – Esecuzioni

- Intuizione. Vogliamo formalizzare l'idea di una macchina che si sposta attraverso le sue configurazioni seguendo le regole dettate dalla sua funzione di transizione
- Sia  $M = \langle \Sigma, \Gamma, Q, \delta, q_0, q_{ves}, q_{no} \rangle$  una macchina di Turing
- Siano  $x, y, z \in \Gamma$  (simboli di nastro)
- Siano  $\sigma, \tau \in \Gamma^*$  (stringhe di simboli di nastro)
- Siano  $q, q' \in Q$
- Definizione. Una configurazione C genera una configurazione D in M ( $C \Rightarrow_M D$ ) se
  - $C = (\sigma y, q, x\tau), D = (\sigma, q', yz\tau) \in \delta(q, x) = (q', z, \leftarrow) \in \sigma \neq \epsilon$  (movimento a sinistra)
  - $C = (\epsilon, q, x\tau), D = (\epsilon, q', z\tau) \in \delta(q, x) = (q', z, \leftarrow)$  (movimento a sinistra bloccato)
  - $C = (\sigma, q, xy\tau), D = (\sigma z, q', y\tau) \in \delta(q, x) = (q', z, \rightarrow) \in \tau \neq \epsilon$  (movimento a destra)
  - $C = (\sigma, q, \epsilon), D = (\sigma z, q', \epsilon)$  e  $\delta(q, \sqcup) = (q', z, \rightarrow)$  (movimento a destra oltre il nastro corrente)
  - $C = (\sigma, q, x\tau), D = (\sigma, q', z\tau) \in \delta(q, x) = (q', z, -)$  (nessun movimento)

#### Macchina di Turing – Esecuzioni – Esempio

• Sia M =  $< \Sigma$ ,  $\Gamma$ , Q,  $\delta$ ,  $q_0$ ,  $q_{yes}$ ,  $q_{no} > \cos \Sigma = \{a, b, \#\}$ ,  $\Gamma = \Sigma \cup \{\sqcup\}$ ,  $Q = \{q_0, q_{yes}, q_{no}\}$ 

q	σ	$oldsymbol{\delta}(q,\sigma)$
$q_0$	а	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_{yes}, b, -)$
$q_0$	#	$(q_{no}, \#, \leftarrow)$
$q_0$	Ц	$(q_0,\sqcup,\to)$

- Esempio. La configurazione iniziale  $C_1 = (\epsilon, q_0, aab)$  genera la configurazione  $C_2 = (a, q_0, ab)$
- Esempio. La configurazione  $C_2 = (a, q_0, ab)$  genera la configurazione  $C_3 = (aa, q_0, b)$
- Esempio. La configurazione  $C_3 = (aa, q_0, b)$  genera la configurazione accettante  $C_4 = (aa, q_{yes}, b)$
- Esempio. La configurazione iniziale  $D_1 = (\epsilon, q_0, aa\#)$  genera la configurazione  $D_2 = (a, q_0, a\#)$
- Esempio. La configurazione  $D_2 = (a, q_0, a\#)$  genera la configurazione  $D_3 = (aa, q_0, \#)$
- Esempio. La configurazione  $D_3 = (aa, q_0, \#)$  genera la configurazione rifiutante  $D_4 = (a, q_{no}, a\#)$

## Macchina di Turing – Linguaggio Riconosciuti

- Definiamo ora il linguaggio riconosciuto da una Macchina di Turing
  - In maniera analoga al linguaggio riconosciuto da un automa
- Definizione. La macchina M accetta\riffiuta l'input  $\sigma \in \Sigma^*$  se esiste una sequenza finita di configurazioni  $C_1, C_2, ..., C_n$  di M tale che le seguenti proprietà sono soddisfatte
  - $C_1 = (\epsilon, q_0, \sigma)$
  - $C_n$  è una configurazione accettante\rifiutante
  - $C_i \Rightarrow_M C_{i+1}$  per ogni i = 1, ..., n-1
- Definizione. L'insieme L(M) delle stringhe che M accetta è detto il linguaggio riconosciuto da M

#### MdT – Linguaggio Riconosciuti – Esempio

• Sia M =  $< \Sigma$ ,  $\Gamma$ , Q,  $\delta$ ,  $q_0$ ,  $q_{yes}$ ,  $q_{no} > \cos \Sigma = \{a, b, \#\}$ ,  $\Gamma = \Sigma \cup \{\sqcup\}$ ,  $Q = \{q_0, q_{yes}, q_{no}\}$ 

$oldsymbol{q}$	σ	$oldsymbol{\delta(q,\sigma)}$
$q_0$	а	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_{yes}, b, -)$
$q_0$	#	$(q_{no}, \#, \leftarrow)$
$q_0$	Ц	$(q_0,\sqcup,\to)$

- Esempio 1. La macchina M accetta l'input  $aab \in \Sigma^*$  a causa di  $(C_1, C_2, C_3, C_4)$ 
  - La configurazione iniziale  $C_1 = (\epsilon, q_0, aab)$  genera la configurazione  $C_2 = (a, q_0, ab)$
  - La configurazione  $C_2 = (a, q_0, ab)$  genera la configurazione  $C_3 = (aa, q_0, b)$
  - Esempio. La configurazione  $C_3 = (aa, q_0, b)$  genera la configurazione accettante  $C_4 = (aa, q_{yes}, b)$

#### MdT – Linguaggio Riconosciuti – Esempio

• Sia M =  $< \Sigma$ ,  $\Gamma$ , Q,  $\delta$ ,  $q_0$ ,  $q_{yes}$ ,  $q_{no} > \cos \Sigma = \{a, b, \#\}$ ,  $\Gamma = \Sigma \cup \{\sqcup\}$ ,  $Q = \{q_0, q_{yes}, q_{no}\}$ 

q	$\sigma$	$oldsymbol{\delta}(oldsymbol{q},oldsymbol{\sigma})$
$q_0$	a	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_{yes}, b, -)$
$q_0$	#	$(q_{no}, \#, \leftarrow)$
$q_0$	Ц	$(q_0,\sqcup,\to)$

- Esempio 2. La macchina M rifiuta l'input  $aa \# \in \Sigma^*$  a causa di  $(D_1, D_2, D_3, D_4)$ 
  - La configurazione iniziale  $D_1 = (\epsilon, q_0, aa\#)$  genera la configurazione  $D_2 = (a, q_0, a\#)$
  - La configurazione  $D_2 = (a, q_0, a\#)$  genera la configurazione  $D_3 = (aa, q_0, \#)$
  - La configurazione  $D_3 = (aa, q_0, \#)$  genera la configurazione rifiutante  $D_4 = (a, q_{no}, a\#)$
- Esempio. Il linguaggio riconosciuto da M è il linguaggio delle stringhe sull'alfabeto  $\{a, b, \#\}$  in cui almeno un simbolo b precede la prima occorrenza del simbolo #

#### MdT – Linguaggio Riconosciuti – Esempio

• Sia M =  $< \Sigma, \Gamma, Q, \delta, q_0, q_{yes}, q_{no} > \cos \Sigma = \{a, b, \#\}, \Gamma = \Sigma \cup \{\sqcup\}, \ Q = \{q_0, q_{yes}, q_{no}\}$ 

$oldsymbol{q}$	σ	$oldsymbol{\delta(q,\sigma)}$
$q_0$	а	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_{yes}, b, -)$
$q_0$	#	$(q_{no}, \#, \leftarrow)$
$q_0$	Ц	$(q_0,\sqcup,\to)$

• Esempio 3. Il linguaggio riconosciuto da M L(M) è il linguaggio delle stringhe sull'alfabeto  $\{a, b, \#\}$  in cui almeno un simbolo b precede la prima occorrenza del simbolo #

## Macchina di Turing – Linguaggio Riconosciuti

- **Domanda**. Data una qualunque stringa  $\sigma \in \Sigma^*$  è vero che una macchina di Turing accetta o rifiuta tale stringa?
  - Considerare anche la macchina che abbiamo appena definito
- Sia M =<  $\Sigma$ ,  $\Gamma$ , Q,  $\delta$ ,  $q_0$ ,  $q_{yes}$ ,  $q_{no}$  > con  $\Sigma$  = {a, b, #},  $\Gamma$  =  $\Sigma$   $\cup$  { $\sqcup$ }, Q = { $q_0$ ,  $q_{yes}$ ,  $q_{no}$ }

q	σ	$oldsymbol{\delta(q,\sigma)}$
$q_0$	а	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_{yes}, b, -)$
$q_0$	#	$(q_{no}, \#, \leftarrow)$
$q_0$	Ц	$(q_0,\sqcup,\to)$

#### Macchina di Turing – Non Terminazione

- **Domanda**. Data una stringa  $\sigma \in \Sigma^*$  è vero che una macchina di Turing accetta o rifiuta tale stringa?
- Sia M =  $< \Sigma$ ,  $\Gamma$ , Q,  $\delta$ ,  $q_0$ ,  $q_{yes}$ ,  $q_{no} > \cos \Sigma = \{a, b, \#\}$ ,  $\Gamma = \Sigma \cup \{\sqcup\}$ ,  $Q = \{q_0, q_{yes}, q_{no}\}$

$\boldsymbol{q}$	σ	$oldsymbol{\delta}(oldsymbol{q},oldsymbol{\sigma})$
$q_0$	a	$(q_0, a, \rightarrow)$
$q_0$	b	$(q_{yes}, b, -)$
$q_0$	#	$(q_{no}, \#, \leftarrow)$
$q_0$	Ц	$(q_0,\sqcup,\to)$

- Esempio. La configurazione iniziale  $C_1 = (\epsilon, q_0, aa)$  genera la configurazione  $C_2 = (a, q_0, a)$
- Esempio. La configurazione  $C_2 = (a, q_0, a)$  genera la configurazione  $C_3 = (aa, q_0, \epsilon)$
- Esempio. La configurazione  $C_3 = (aa, q_0, \epsilon)$  genera la configurazione  $C_4 = (aa \sqcup, q_0, \epsilon)$
- Esempio. La configurazione  $C_5 = (aa \sqcup, q_0, \epsilon)$  genera la configurazione  $C_6 = (aa \sqcup \sqcup, q_0, \epsilon)$
- •
- Possiamo conclude che M non accetta ne rifiuta l'input aa!!
  - Intuitivamente, la macchina entra in un ciclo infinito

#### Macchina di Turing – Terminazione

- Sia  $M = \langle \Sigma, \Gamma, Q, \delta, q_0, q_{ves}, q_{no} \rangle$  una macchina di Turing
- Definizione 1. La macchina M è terminante se per ogni stringa  $\sigma \in \Sigma^*$  abbiamo che M accetta o rifiuta  $\sigma$ 
  - La macchina entra nello stato accettante o rifiutante per ogni input senza mai bloccarsi in un ciclo infinito
  - Tali macchine vengono chiamate a volte decisori
- Definizione 2. Un linguaggio di stringhe  $\mathcal{L}$  è detto Turing Riconoscibile se esiste una macchina di Turing M che riconosce  $\mathcal{L}$  (ovvero  $\mathcal{L} = L(M)$ )
  - Spesso definiti Ricorsivamente Enumerabili (Recursively Enumerable)
- Definizione 3. Un linguaggio di stringhe  $\mathcal{L}$  è detto Turing Decidibile se esiste una macchina di Turing terminante M che riconosce  $\mathcal{L}$  (ovvero  $\mathcal{L} = L(M)$ )
  - Spesso definiti Ricorsivi (Recursive)
- **Domanda 1**. Esiste un linguaggio di stringhe £ che non è Turing Riconoscibile?
- Domanda 2. Esiste un linguaggio di stringhe  $\mathcal{L}'$  che non è Turing Decidibile?

#### Macchina di Turing – Decidibilità

- Teorema 1. Esiste un linguaggio di stringhe  $\mathcal{L}$  che non è Turing Riconoscibile
- Teorema 2. Esiste un linguaggio di stringhe  $\mathcal{L}'$  che non è Turing Decidibile
- Le prove di questi due teoremi ci terranno impegnati per le prossime lezioni ©

# Esempi di Macchina di Turing

#### Algoritmo per riconoscere $L = \{a^m b^m c^m | m \ge 0\}$

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- 6. Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4. Torna** all'ultima occorrenza del carattere ↓



Stato 0.

La macchina esamina il simbolo corrente e verifica le condizioni di terminazione

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4. Torna** all'ultima occorrenza del carattere ↓

Stato 1.

La macchina cerca il prossimo carattere *b* e lo sovrascrive se lo trova oppure rifiuta se trova caratteri non coerenti

Algoritmo per riconoscere  $L = \{a^m b^m c^m | m \ge 0\}$ 

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- 6. Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere *b* 
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - 1. Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

Stato 2.

La macchina cerca il prossimo carattere *c* e lo sovrascrive se lo trova oppure rifiuta



#### Algoritmo per riconoscere $L = \{a^m b^m c^m | m \ge 0\}$

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- 6. Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la strin
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4. Torna** all'ultima occorrenza del carattere ↓

#### Stato 3.

brrente

La macchina torna indietro all'ultima occorrenza di ↓ e torna allo Stato 0

- Definiamo la Macchina di Turing  $M = < \Sigma, \Gamma, Q, \delta, q_0, q_{yes}, q_{no} > con$ 
  - $\ \Sigma = \{a,b\}, \Gamma = \Sigma \cup \{\downarrow,\#,\sqcup\}, \ Q = \{q_0,q_1,q_2,q_3,q_{yes},q_{no}\}$

- Procediamo a definire la funzione di transizione  $\delta$  stato per ogni stato
  - Tenendo in considerazione le diverse fasi

- Stato 0. La macchina esamina il simbolo corrente e verifica le condizioni di terminazione
  - Se x = # Allora passa al prossimo carattere.
    - $-\delta(q_0,\#) = (q_0,\#,\to)$
  - Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
    - $\delta(q_0, \sqcup) = (q_{ves}, \sqcup, -)$
  - Se x = b Allora rifiuta la stringa corrente
    - $-\delta(q_0,b) = (q_{no},b,-)$
  - Se x = c Allora rifiuta la stringa corrente
    - $\delta(q_0, c) = (q_{no}, c, -)$
  - Se  $x = \downarrow$  Allora rifiuta la stringa corrente
    - $\delta(q_0,\downarrow) = (q_{no},\downarrow,-)$
  - Se x = a Allora Sovrascrivi tale occorrenza di a con il carattere  $\downarrow$  e passa allo Stato 1
    - $\delta(q_0, a) = (q_1, \downarrow, \rightarrow)$

- Stato 1. La macchina cerca il prossimo carattere b e lo sovrascrive se lo trova oppure rifiuta se trova un carattere indesiderato
  - Se x = # Allora passa al prossimo carattere.
    - $-\delta(q_1,\#) = (q_1,\#,\to)$
  - Se  $x = \sqcup$  (cella vuota) oppure  $x = \downarrow$  Allora rifiuta la stringa corrente
    - $\delta(q_1, \sqcup) = (q_{n_0}, \sqcup, -)$
    - $\delta(q_1,\downarrow) = (q_{no},\downarrow,-)$
  - Se trova a Allora scorri a destra il nastro
    - $\delta(q_1, a) = (q_1, a, \rightarrow)$
  - Se trova b Allora sovrascrivilo con # e passa allo Stato 2
    - $-\delta(q_1,b) = (q_2,\#,\to)$
  - Se trova c Allora rifiuta la stringa corrente
    - $-\delta(q_1,c) = (q_{no},c,-)$

- Stato 2. La macchina cerca il prossimo carattere b e lo sovrascrive se lo trova oppure rifiuta se trova un carattere indesiderato
  - Se x = # Allora passa al prossimo carattere.
    - $-\delta(q_2,\#) = (q_2,\#,\to)$
  - Se  $x = \sqcup$  (cella vuota) oppure  $x = \downarrow$  Allora rifiuta la stringa corrente
    - $\delta(q_2, \sqcup) = (q_{no}, \sqcup, -)$
    - $\delta(q_2,\downarrow) = (q_{no},\downarrow,-)$
  - Se trova b Allora scorri a destra il nastro
    - $\delta(q_2, b) = (q_2, b, \rightarrow)$
  - Se trova c Allora sovrascrivilo con # e passa allo Stato 3
    - $-\delta(q_2,c)=(q_3,\#,-)$
  - Se trova a Allora rifiuta la stringa corrente
    - $\delta(q_2, a) = (q_{no}, a, -)$

- Stato 3. La macchina torna indietro all'ultima occorrenza di ↓ e torna allo Stato 0
  - Se trova ↓ Allora passa allo Stato 0
    - $\delta(q_3,\downarrow) = (q_0,\downarrow,\rightarrow)$
  - - $\delta(q_3, \sqcup) = (q_{no}, \sqcup, -)$
  - Altrimenti scorre a sinistra
    - $\delta(q_3, a) = (q_3, a, \leftarrow)$
    - $-\delta(q_3,b)=(q_3,b,\leftarrow)$
    - $\delta(q_3, c) = (q_3, c, \leftarrow)$
    - $\delta(q_3, \#) = (q_3, \#, \leftarrow)$

$q \in Q$	$\sigma \in \Gamma$	$\delta(\sigma,q)$
$q_0$	#	$(q_0, \#, \rightarrow)$
$q_0$	Ц	$(q_{yes},\sqcup,-)$
$q_0$	$\downarrow$	$(q_{no},\downarrow,-)$
$q_0$	а	$(q_1,\downarrow,\rightarrow)$
$q_0$	b	$(q_{no},b,-)$
$q_0$	С	$(q_{no},c,-)$
$q_1$	#	$(q_1, \#, \rightarrow)$
$q_1$	Ц	$(q_{no},\sqcup,-)$
$q_1$	1	$(q_{no},\downarrow,-)$
$q_1$	а	$(q_1, a, \rightarrow)$
$q_1$	b	$(q_2, \#, \rightarrow)$
$q_1$	С	$(q_{no},c,-)$

$q \in Q$	$\sigma \in \Gamma$	$\delta(\sigma,q)$
$q_2$	#	$(q_2, \#, \rightarrow)$
$q_2$	Ц	$(q_{no},\sqcup,-)$
$q_2$	1	$(q_{no},\downarrow,-)$
$q_2$	а	$(q_{no},a,-)$
$q_2$	b	$(q_2, b, \rightarrow)$
$q_2$	С	$(q_3, \#, -)$
$q_3$	#	$(q_3, \#, \leftarrow)$
$q_3$	Ц	$(q_{no},\sqcup,-)$
$q_3$	1	$(q_0,\downarrow,\rightarrow)$
$q_3$	а	$(q_3, a, \leftarrow)$
$q_3$	b	$(q_3, b, \leftarrow)$
$q_3$	С	$(q_3, c, \leftarrow)$

#### Macchine di Turing – Esempio di Computazione

#### Algoritmo per riconoscere $L = \{a^m b^m c^m | m \ge 0\}$

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓

#### **NASTRO**

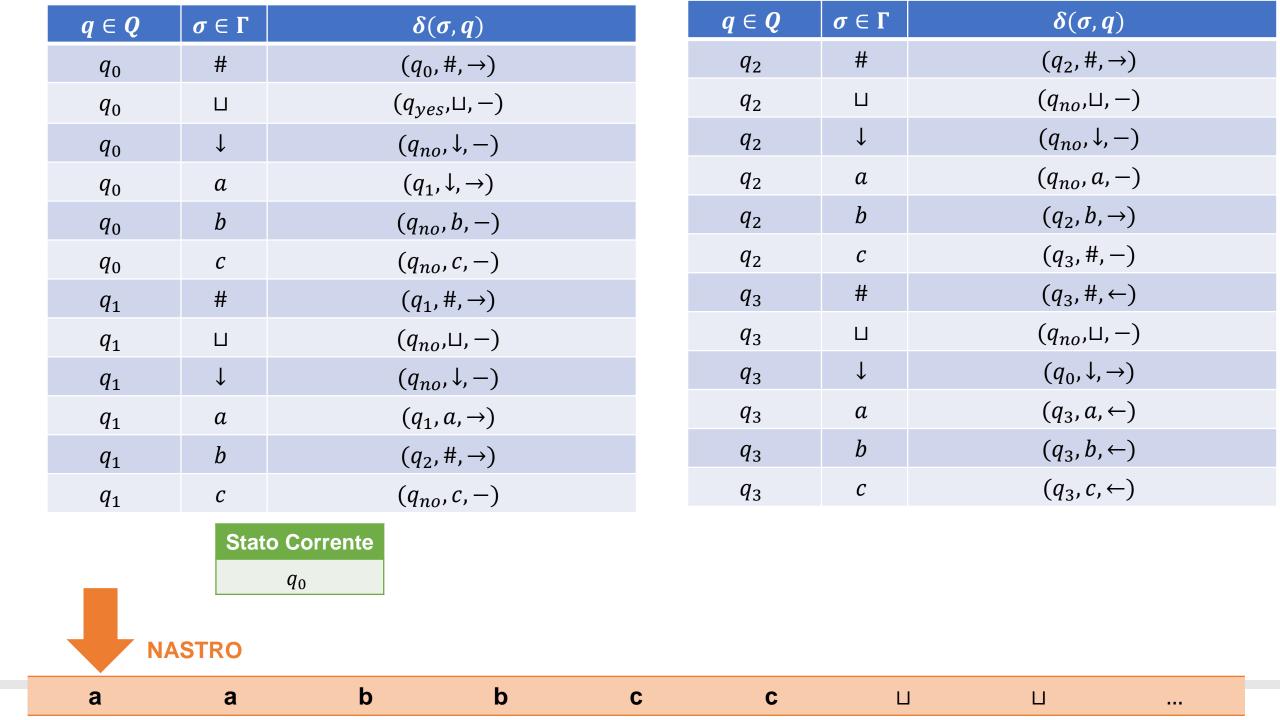
a a b b c c u u ...

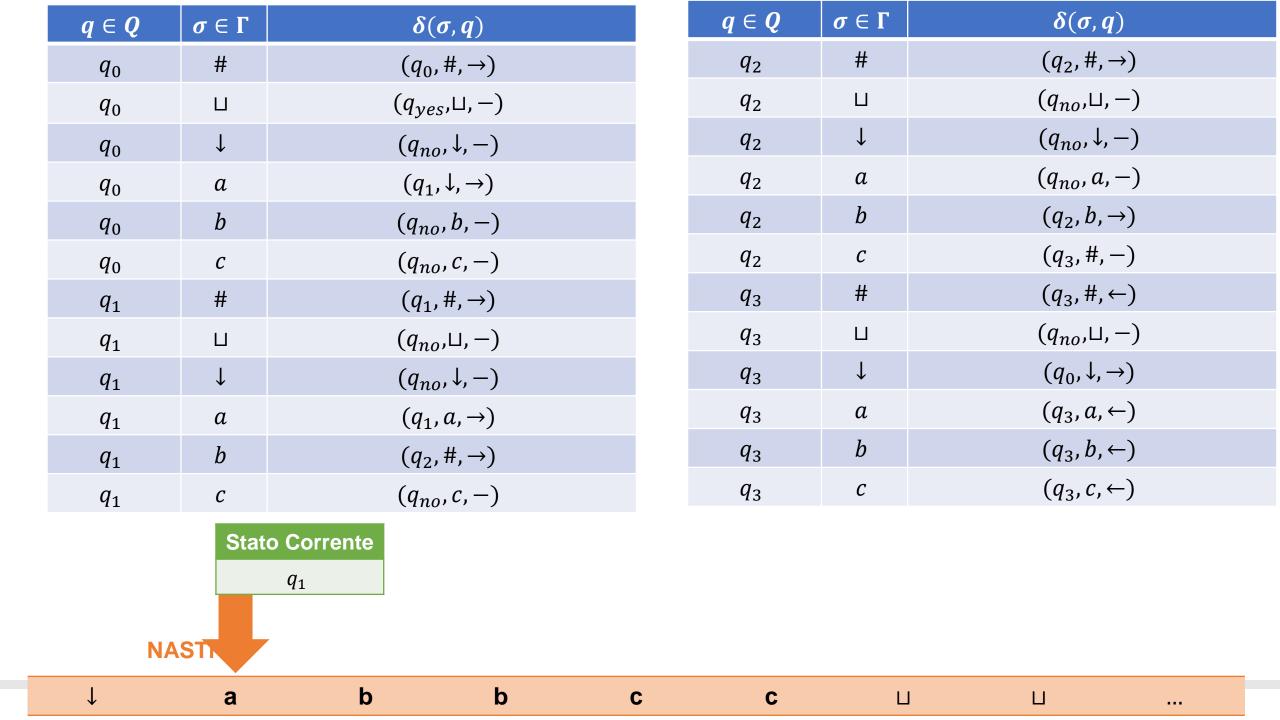
$q \in Q$	$\sigma \in \Gamma$	$oldsymbol{\delta}(oldsymbol{\sigma},oldsymbol{q})$
$q_0$	#	$(q_0, \#, \rightarrow)$
$q_0$	Ц	$(q_{yes}$ , $\sqcup$ , $-)$
$q_0$	$\downarrow$	$(q_{no},\downarrow,-)$
$q_0$	а	$(q_1,\downarrow,\rightarrow)$
$q_0$	b	$(q_{no},b,-)$
$q_0$	С	$(q_{no},c,-)$
$q_1$	#	$(q_1,\#,\to)$
$q_1$	Ш	$(q_{no},\sqcup,-)$
$q_1$	<b>↓</b>	$(q_{no},\downarrow,-)$
$q_1$	а	$(q_1, a, \rightarrow)$
$q_1$	b	$(q_2, \#, \rightarrow)$
$q_1$	С	$(q_{no},c,-)$

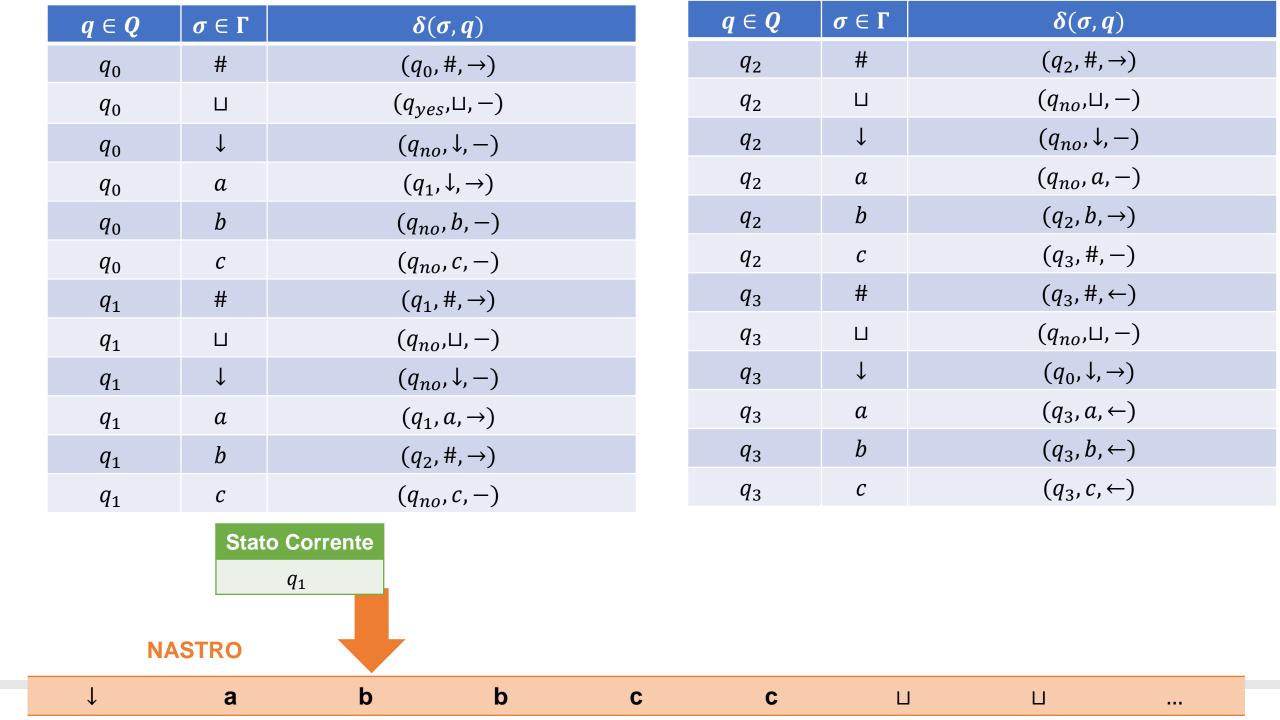
$q \in Q$	$\sigma \in \Gamma$	$\delta(\sigma,q)$
$q_2$	#	$(q_2, \#, \rightarrow)$
$q_2$	Ц	$(q_{no}$ , $\sqcup$ , $-)$
$q_2$	$\downarrow$	$(q_{no},\downarrow,-)$
$q_2$	а	$(q_{no},a,-)$
$q_2$	b	$(q_2, b, \rightarrow)$
$q_2$	С	$(q_3, \#, -)$
$q_3$	#	$(q_3, \#, \leftarrow)$
$q_3$	Ш	$(q_{no},\sqcup,-)$
$q_3$	$\downarrow$	$(q_0,\downarrow,\rightarrow)$
$q_3$	a	$(q_3, a, \leftarrow)$
$q_3$	b	$(q_3, b, \leftarrow)$
$q_3$	С	$(q_3, c, \leftarrow)$

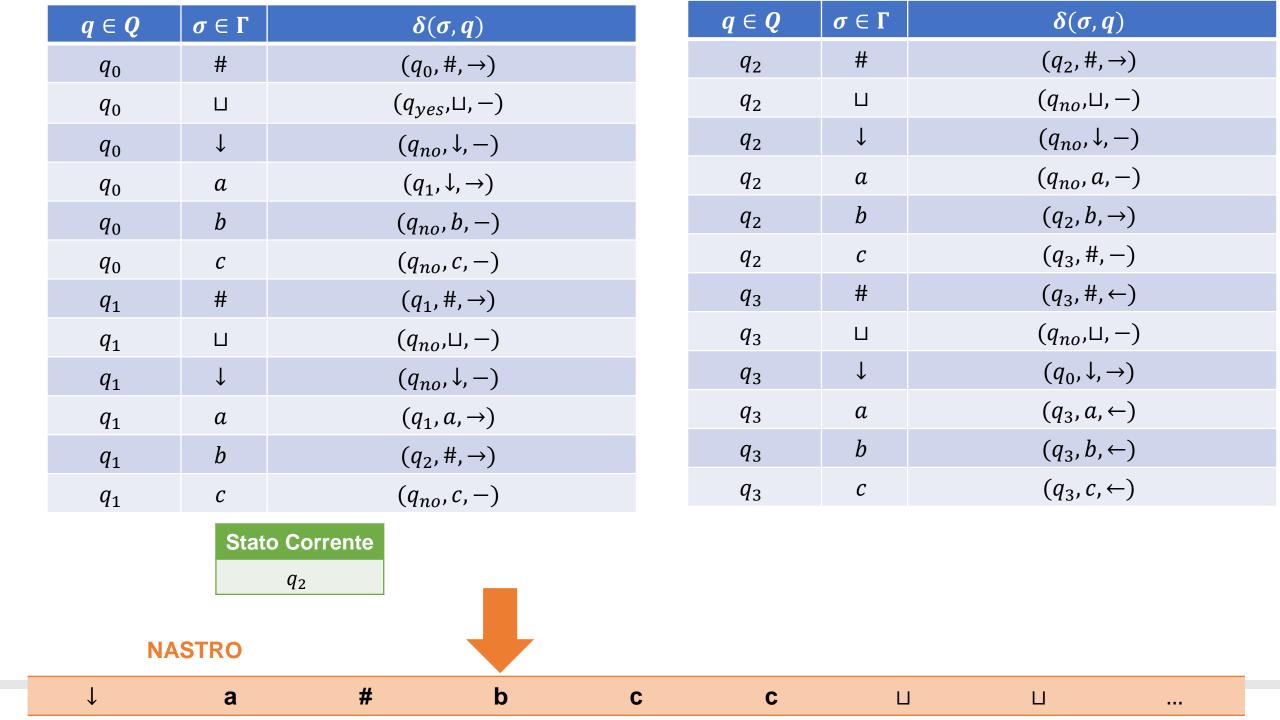
#### **NASTRO**

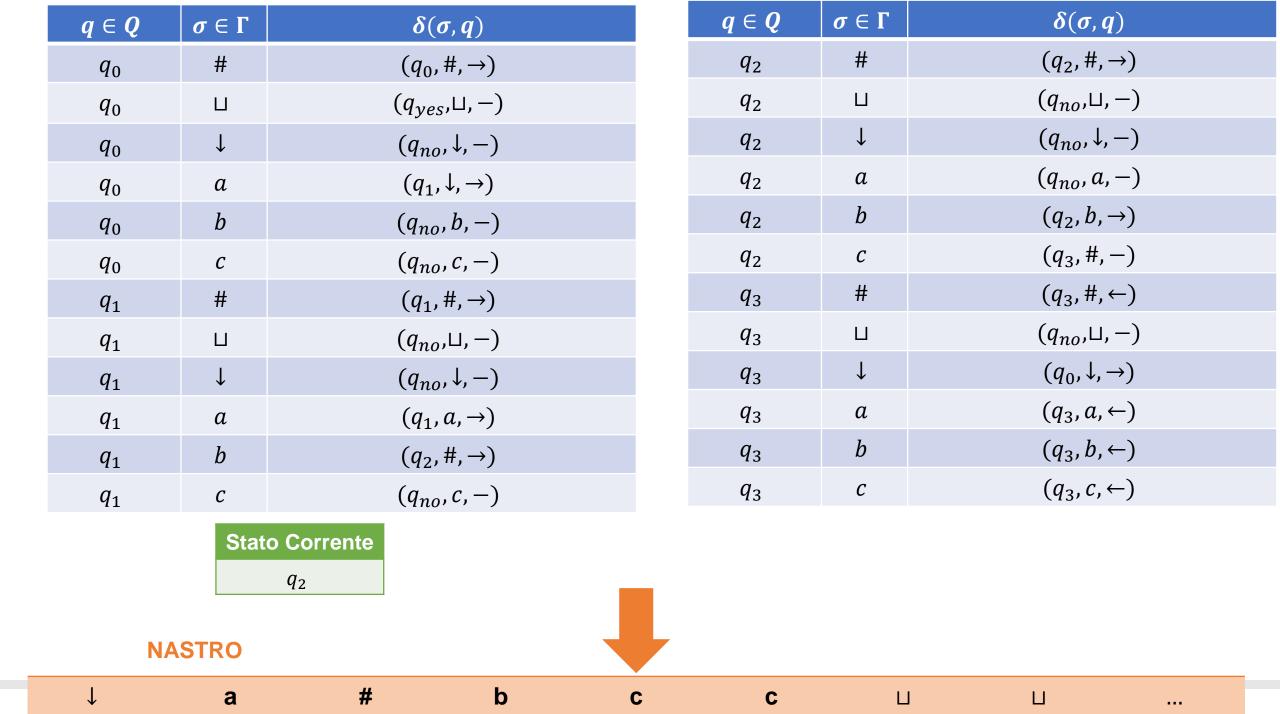
a a b b c c u u ...

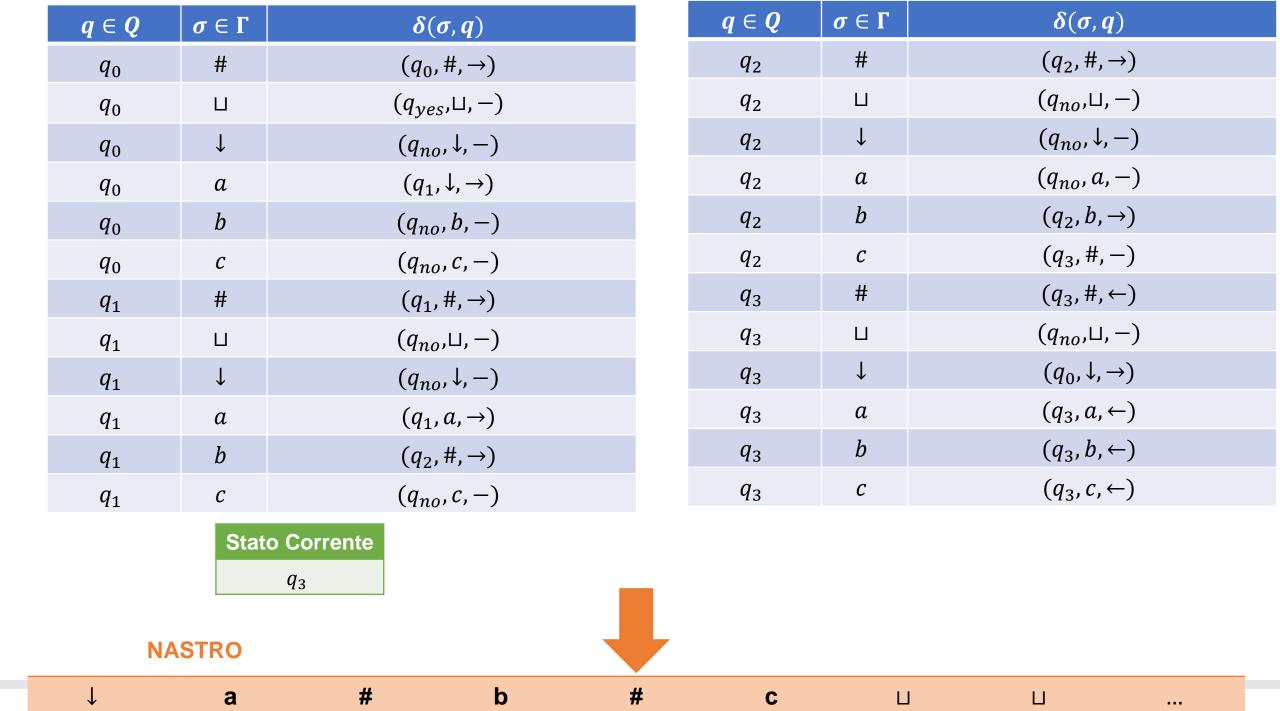


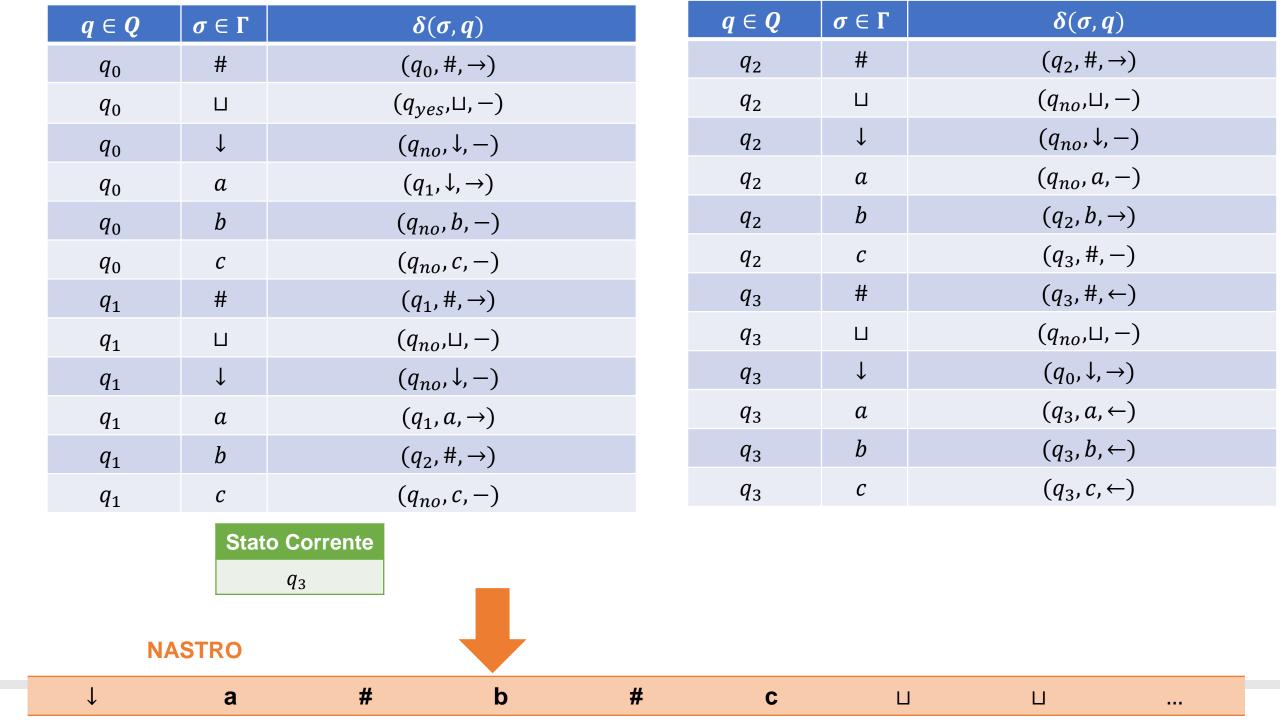


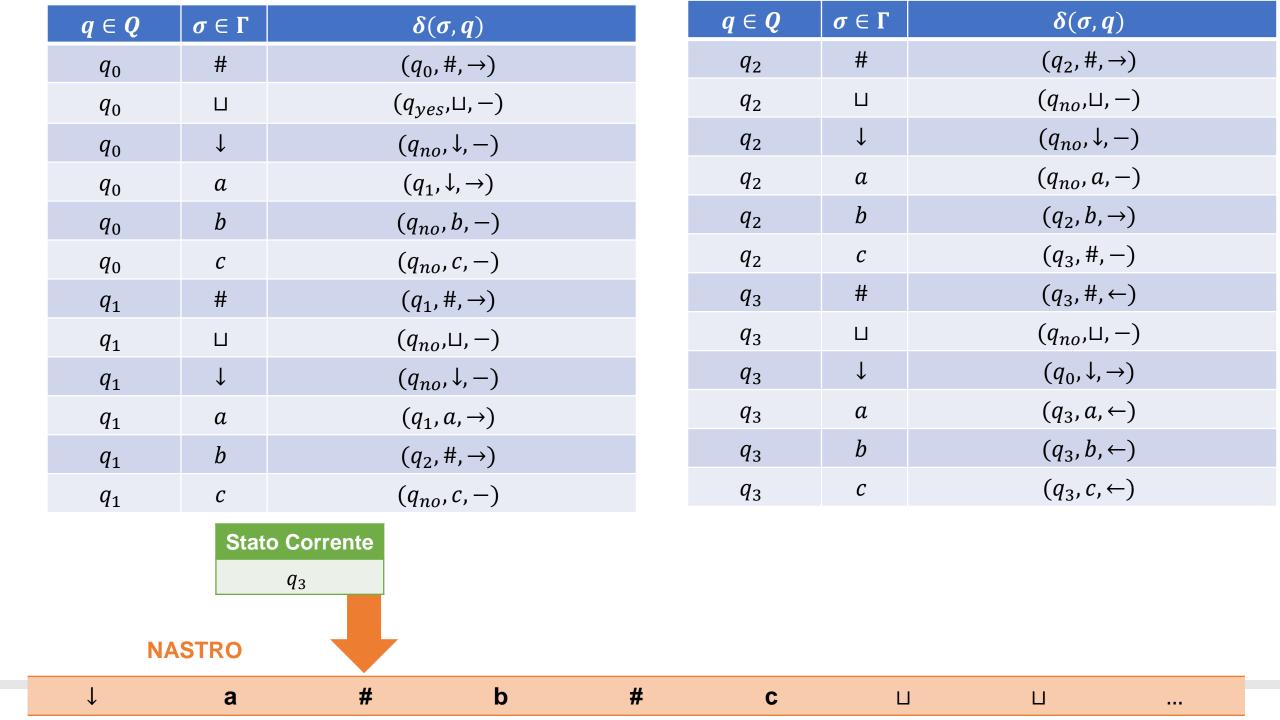


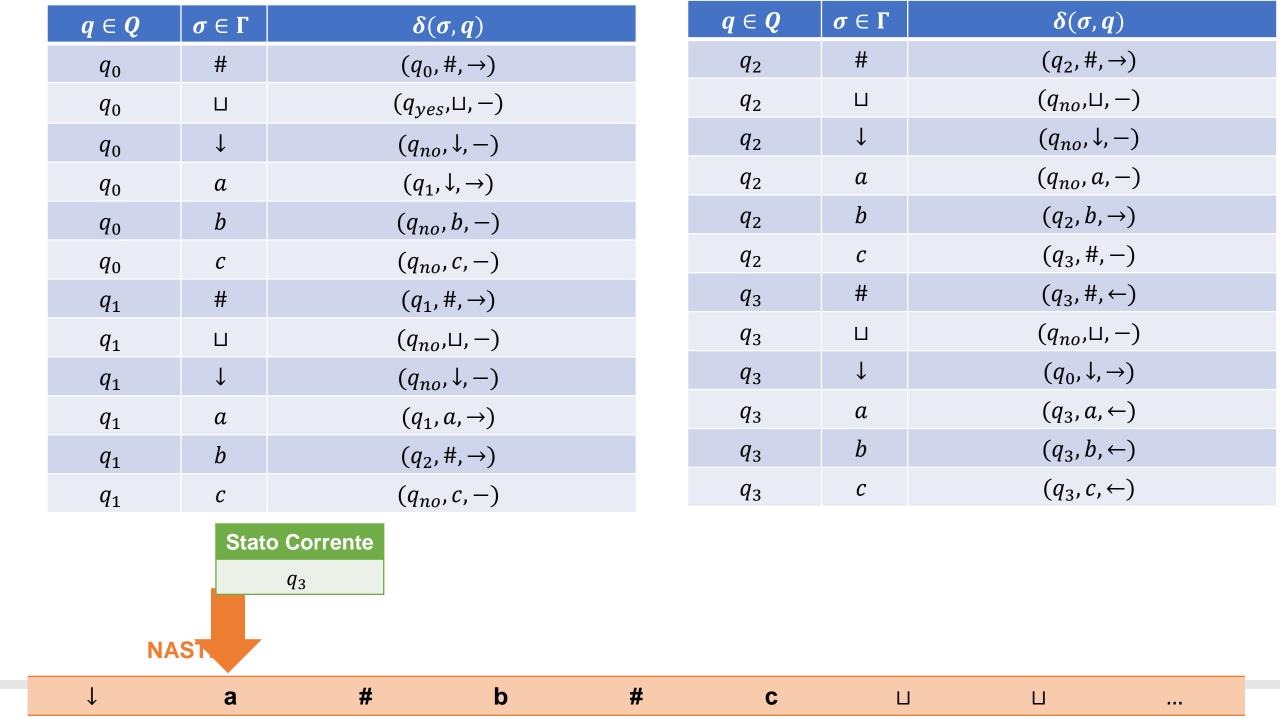


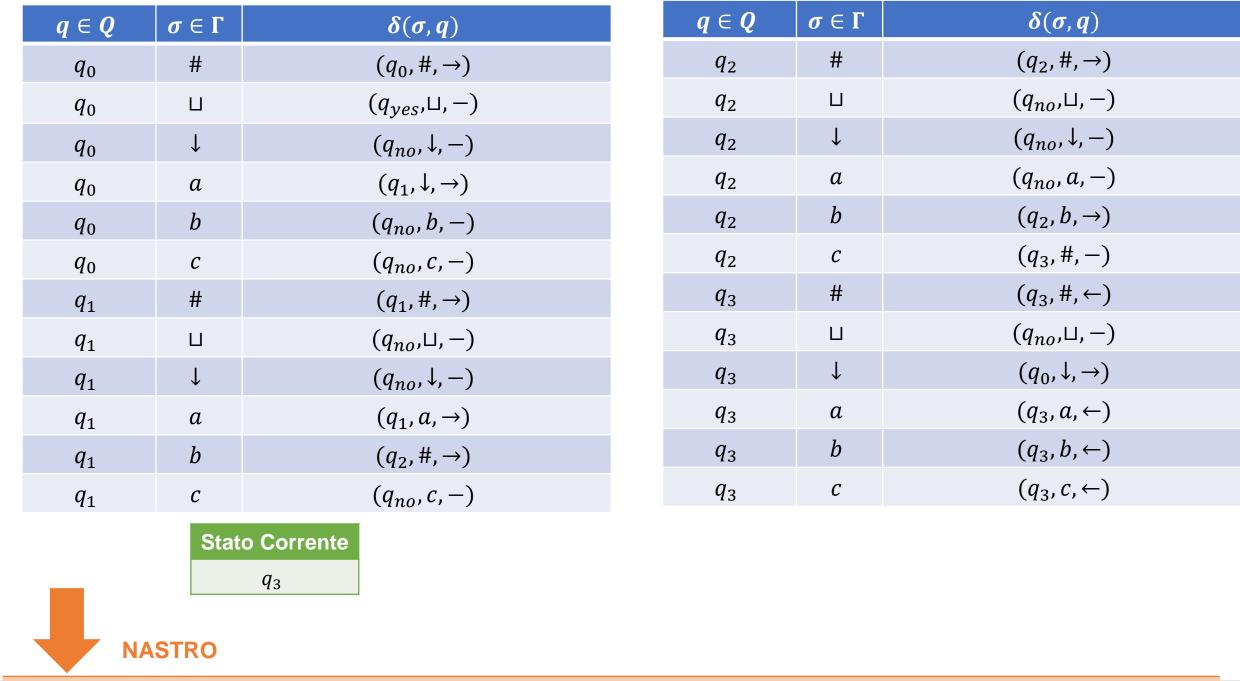




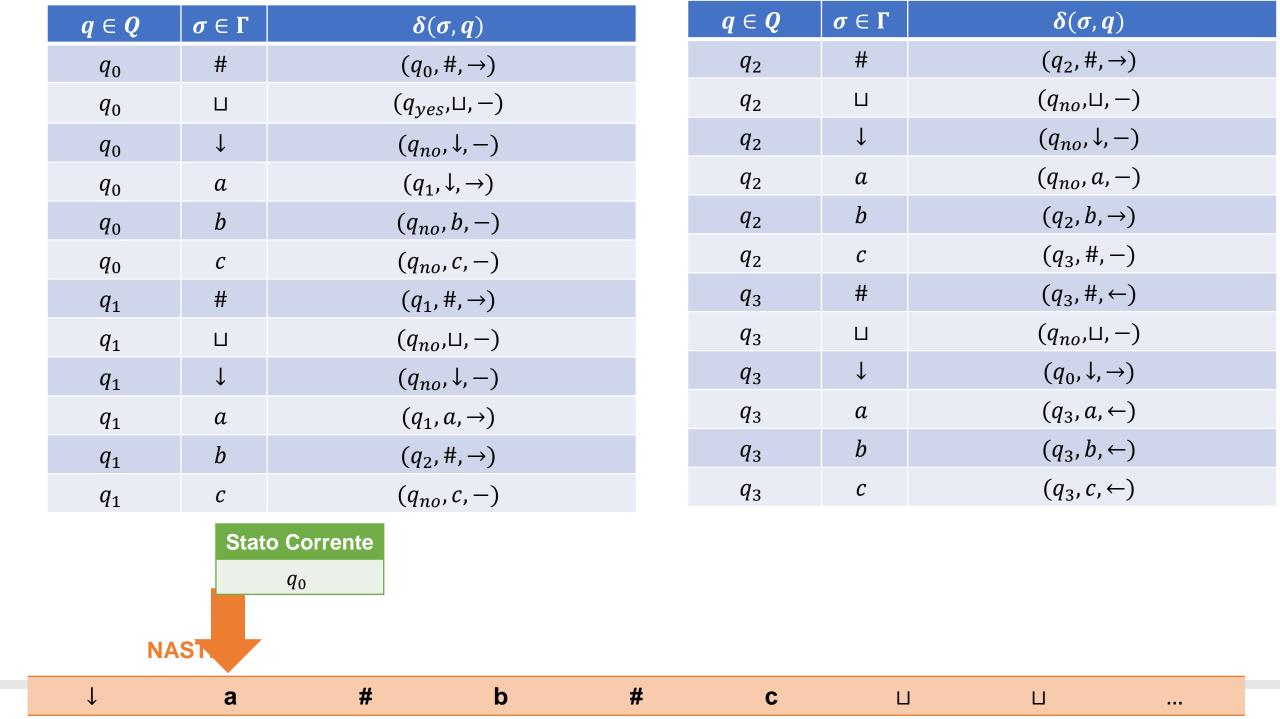


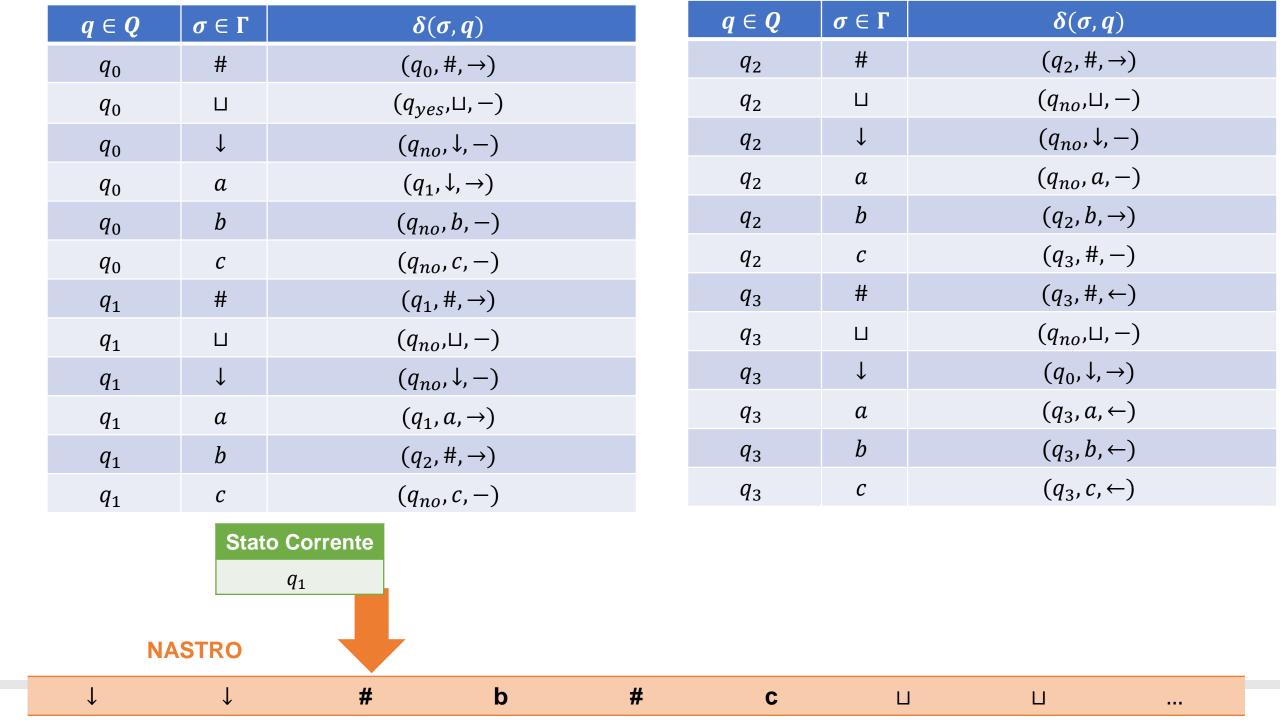


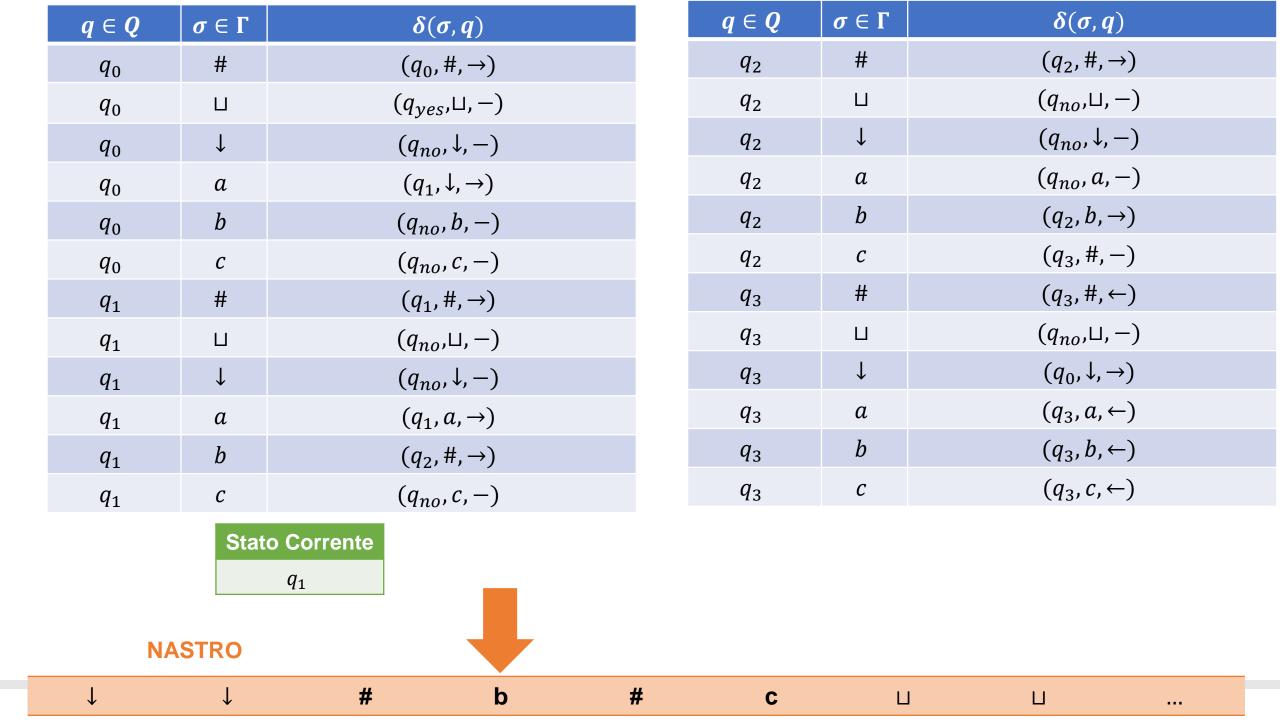


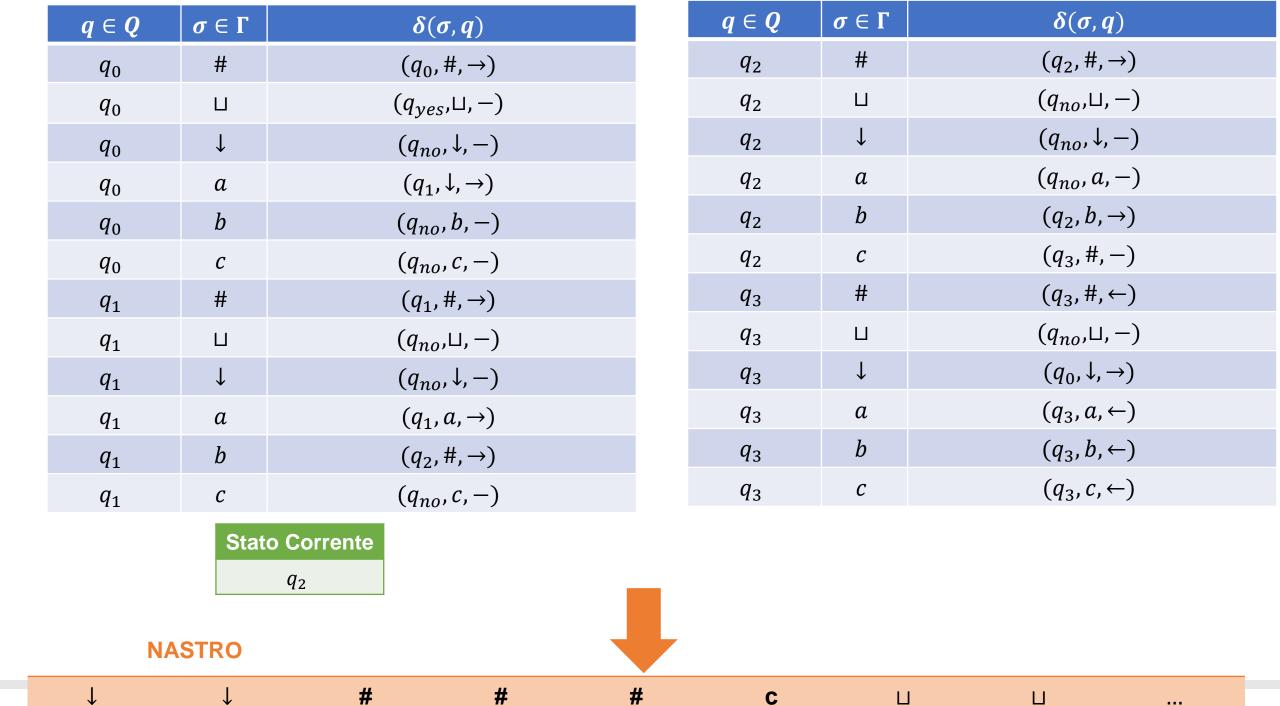


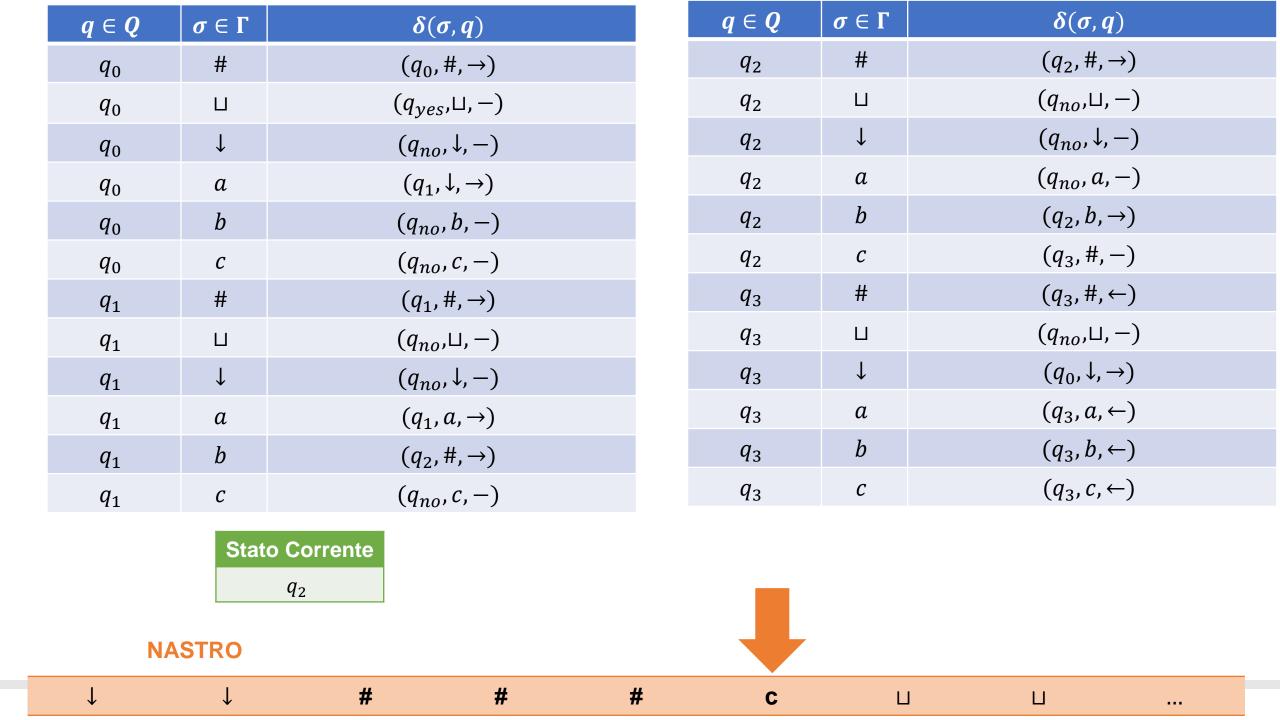
↓ a # b # c ⊔ ⊔ ...

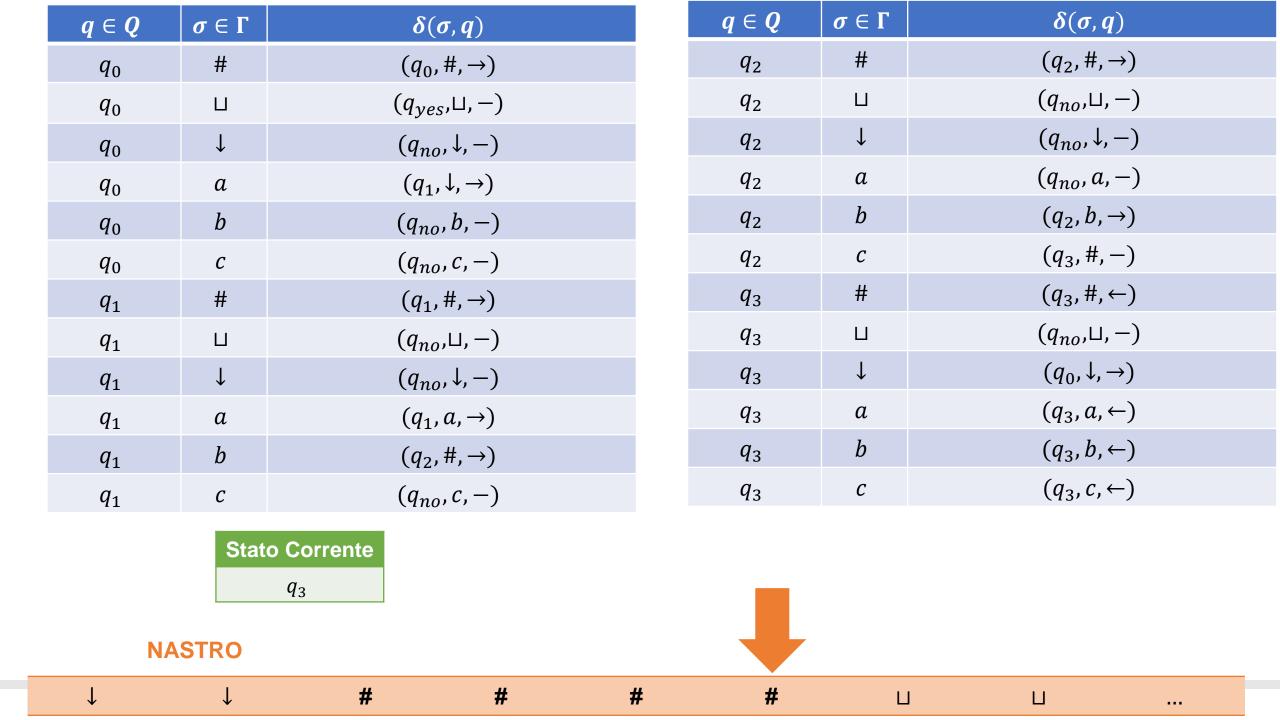


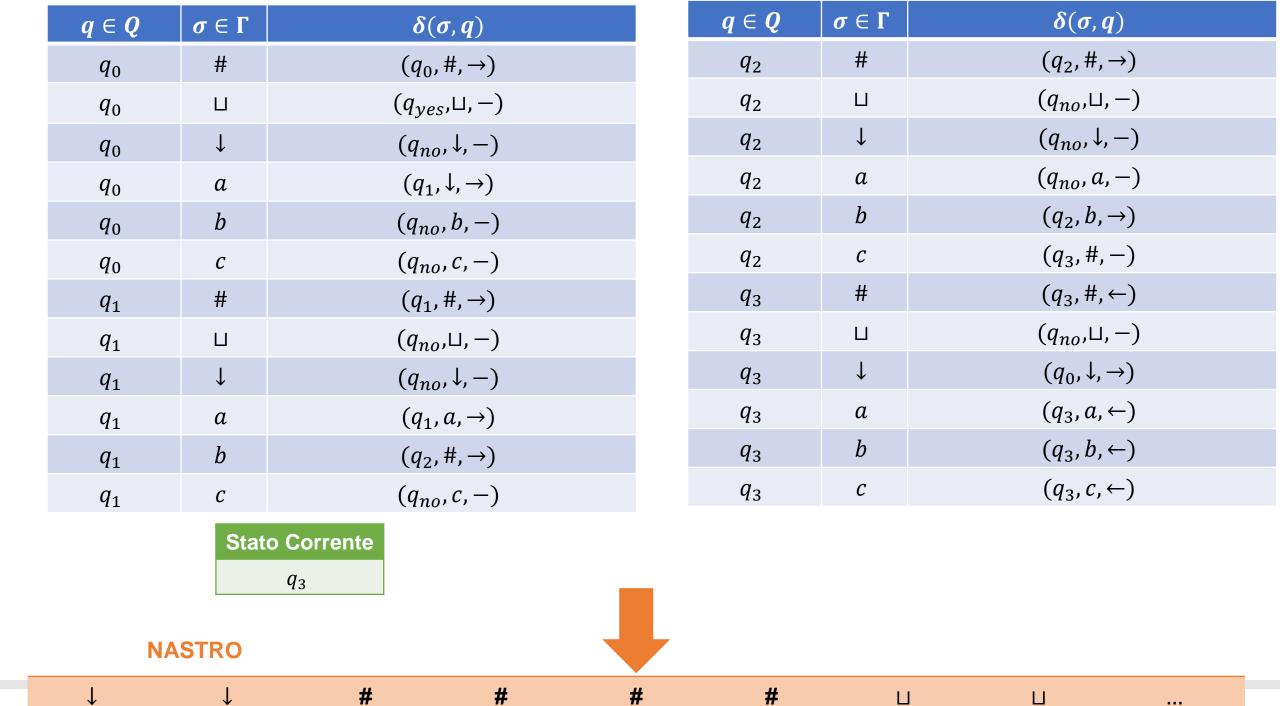


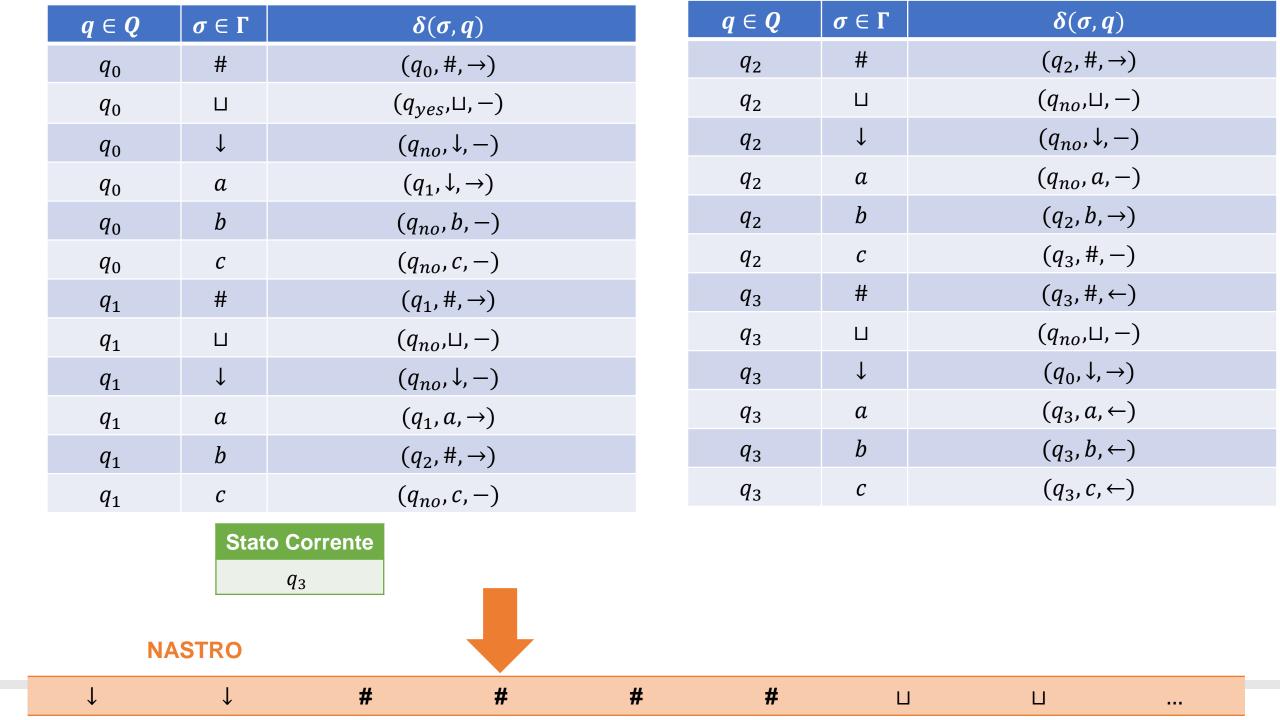


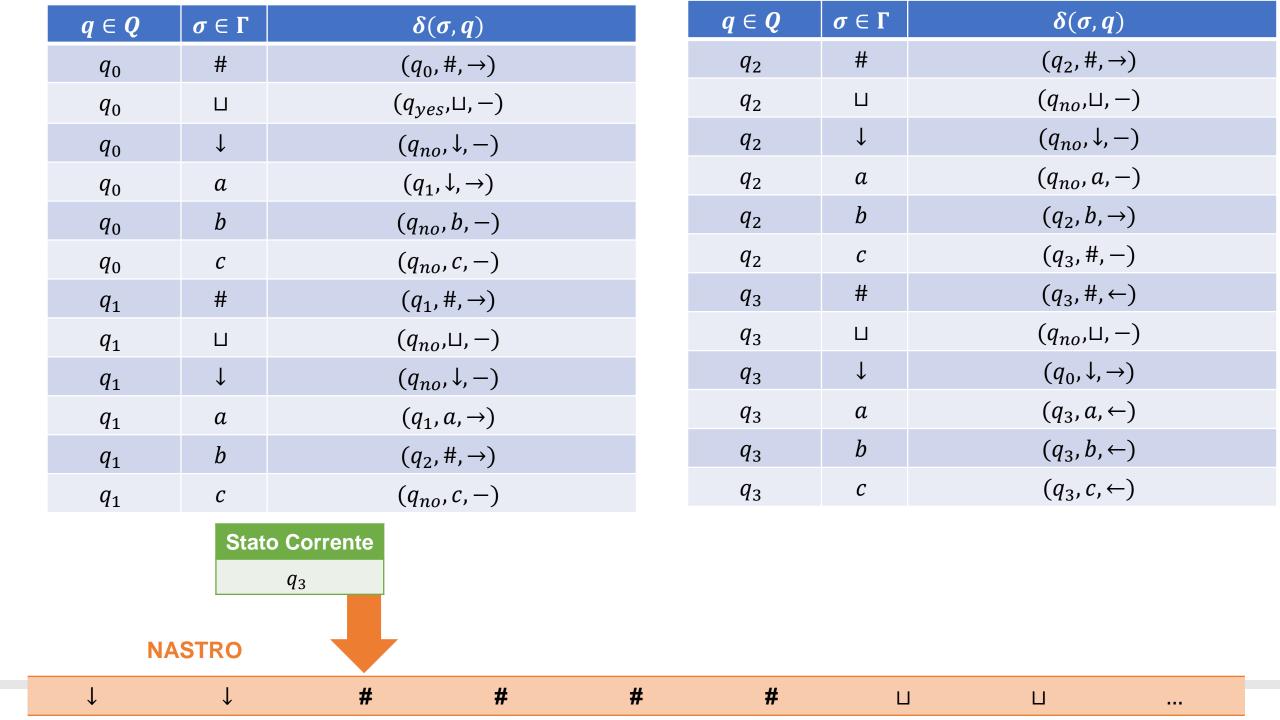


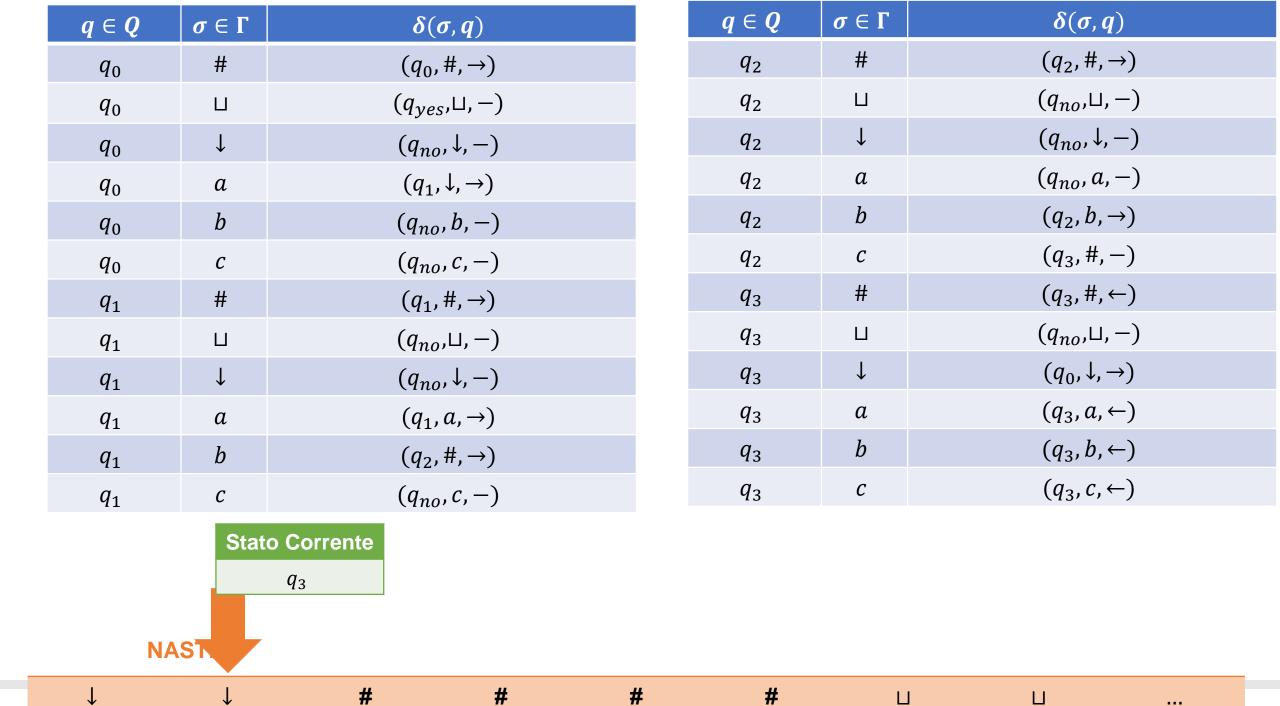


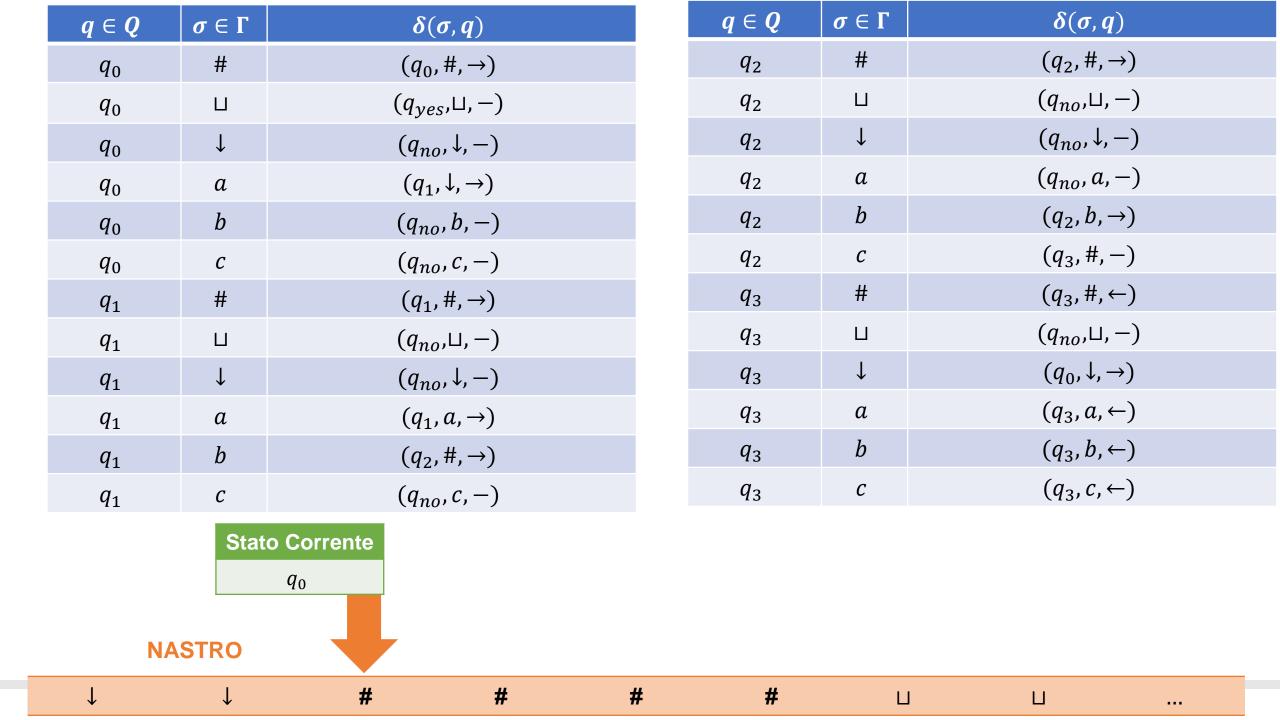


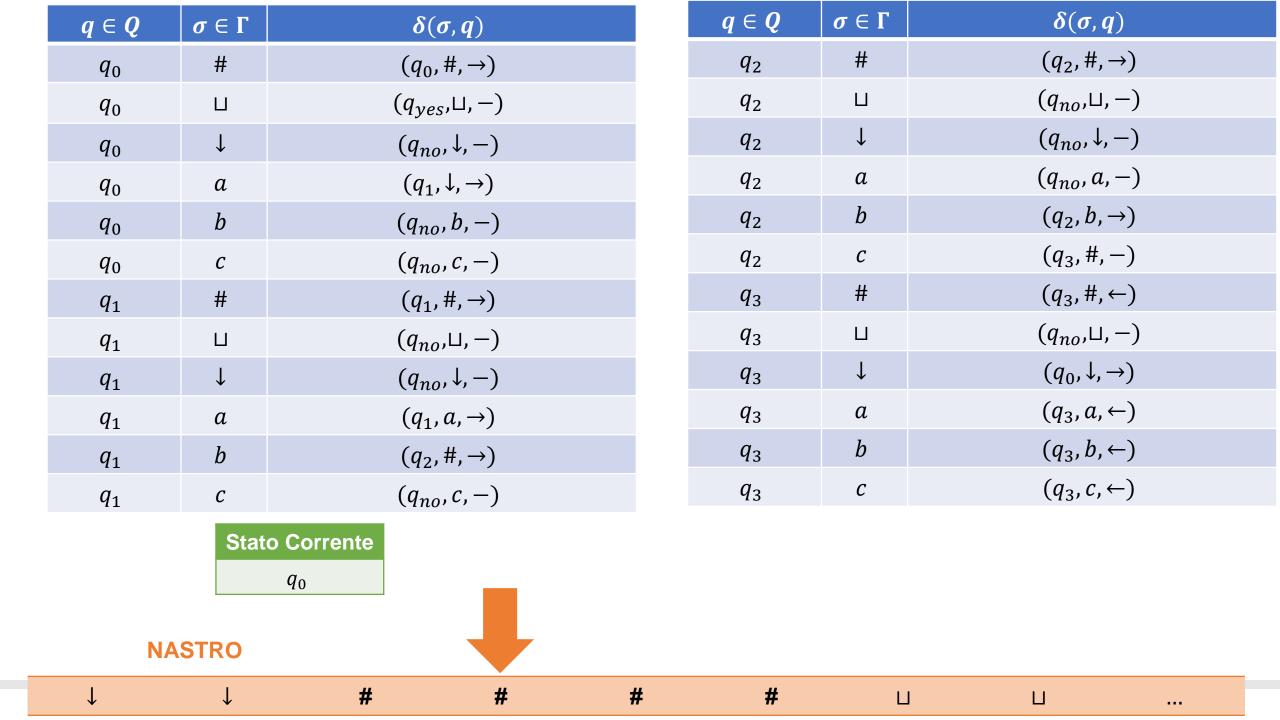


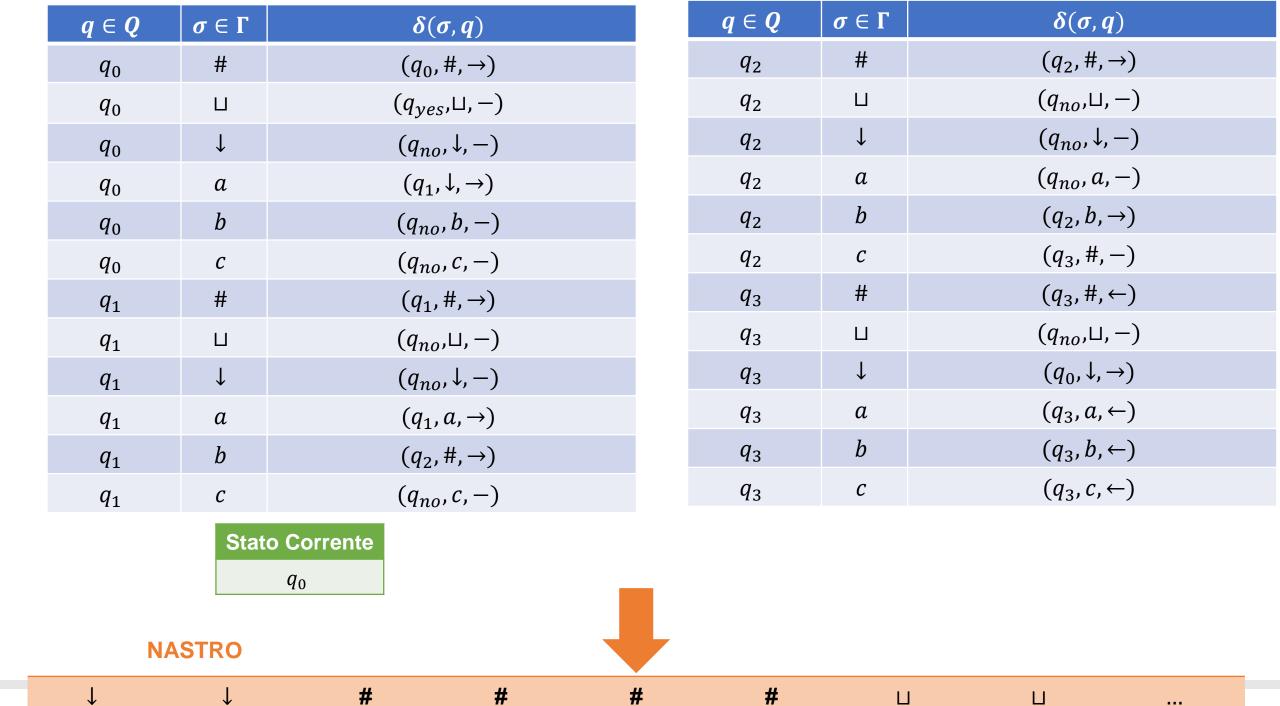


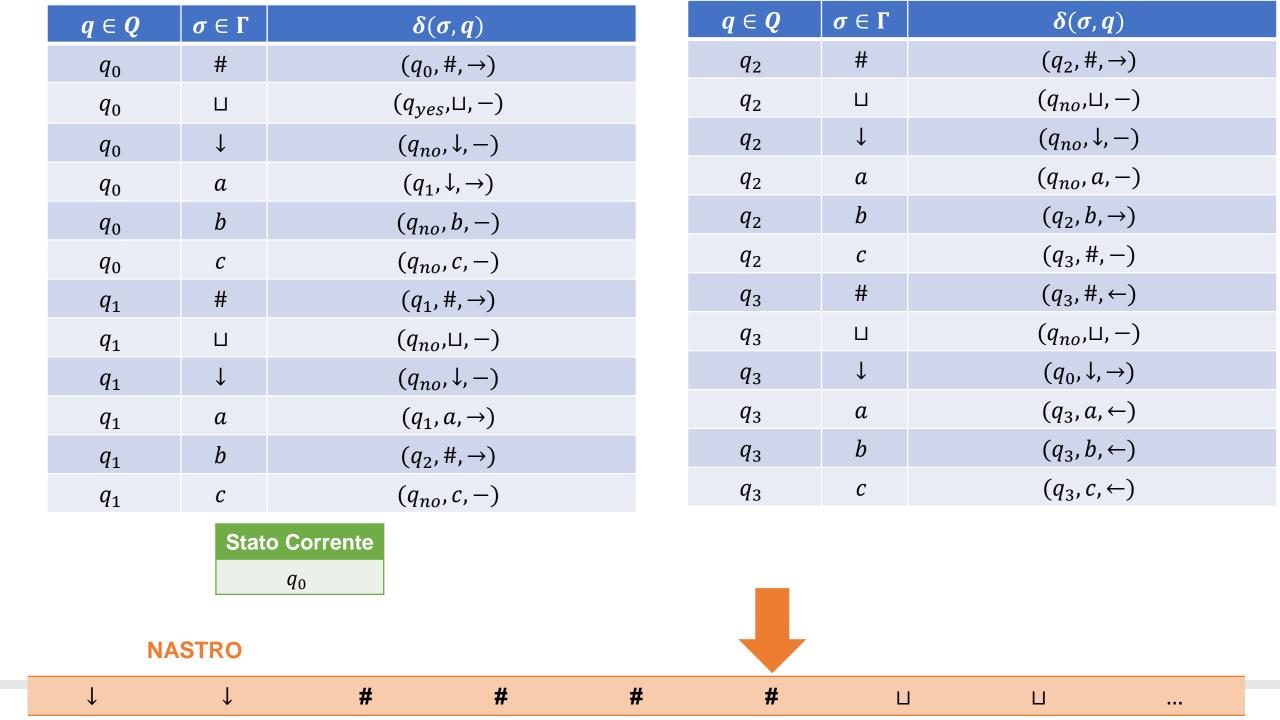


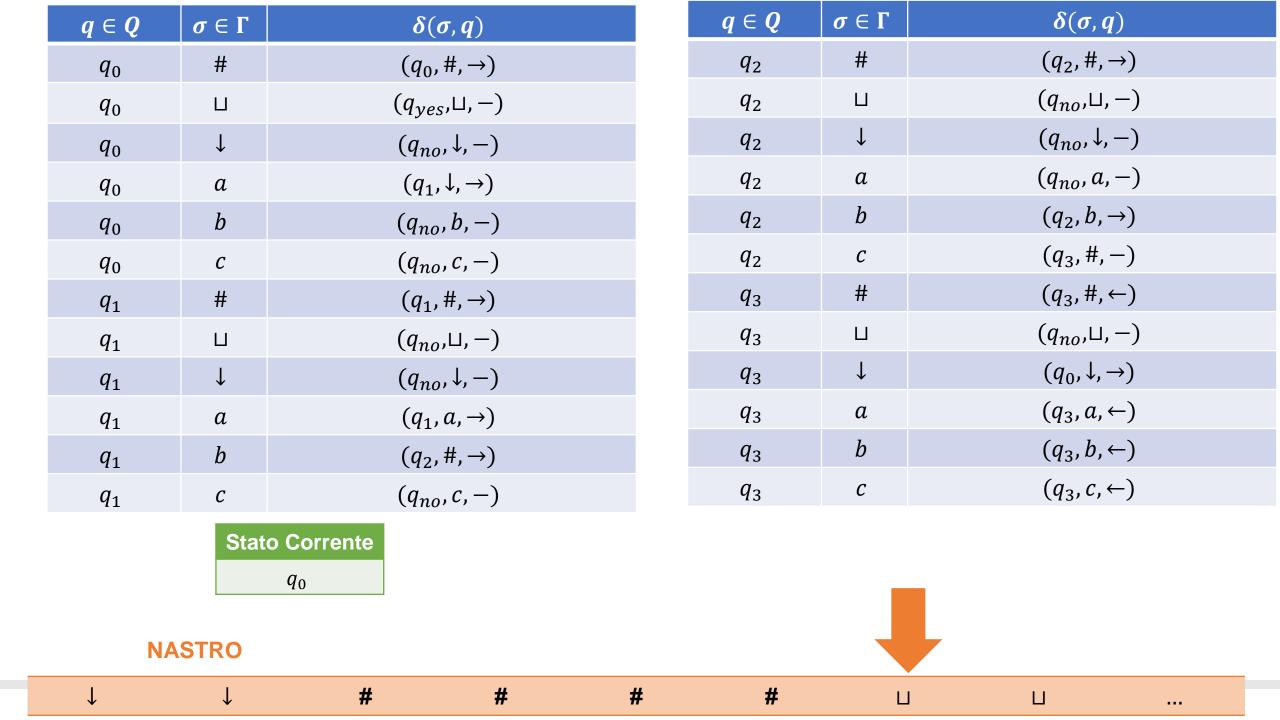


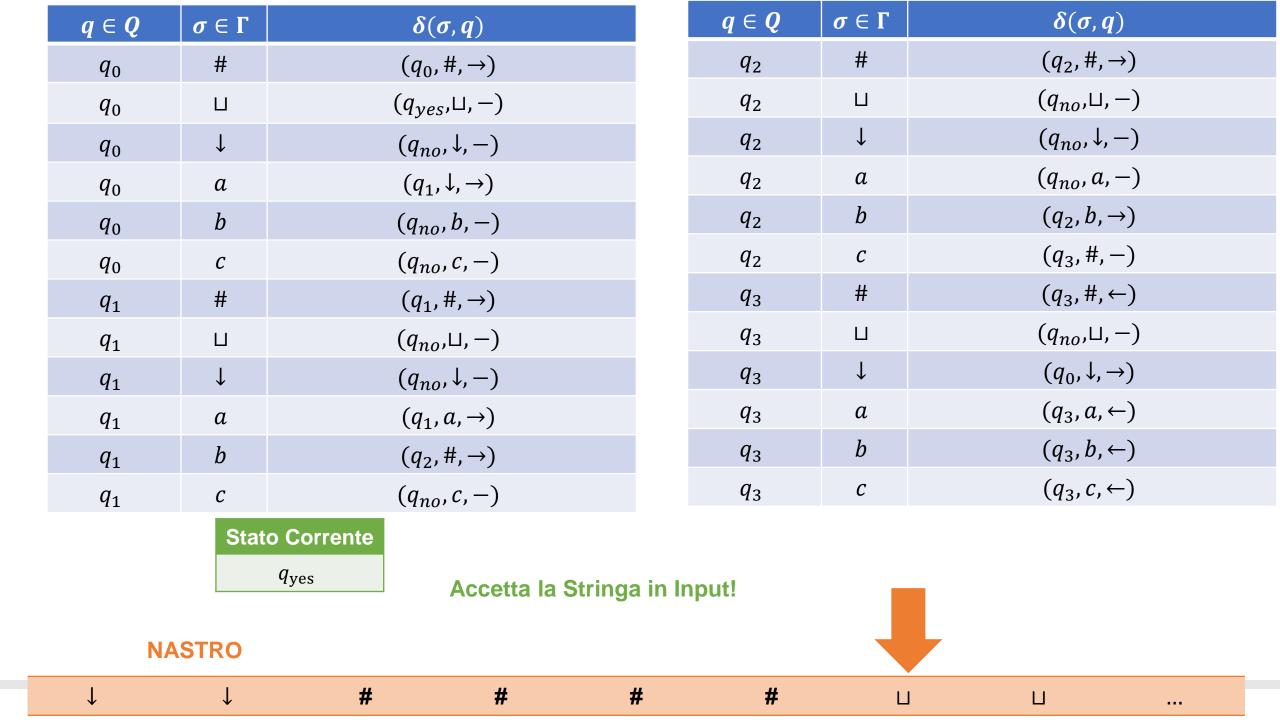










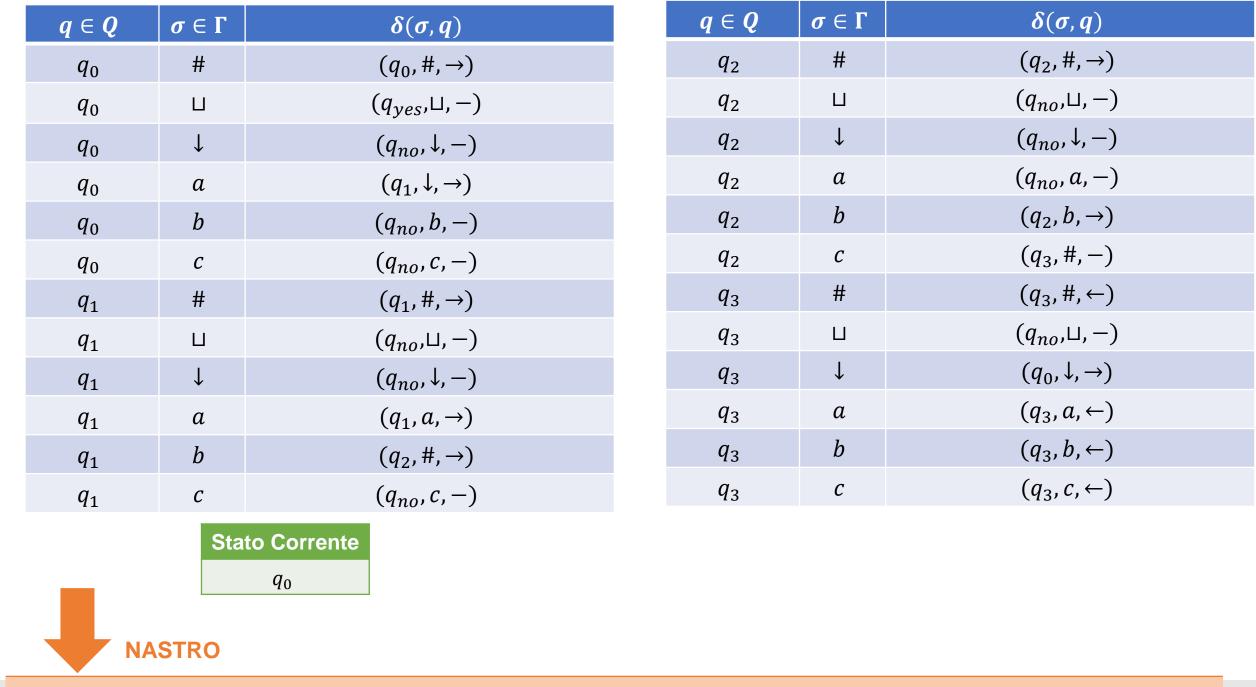


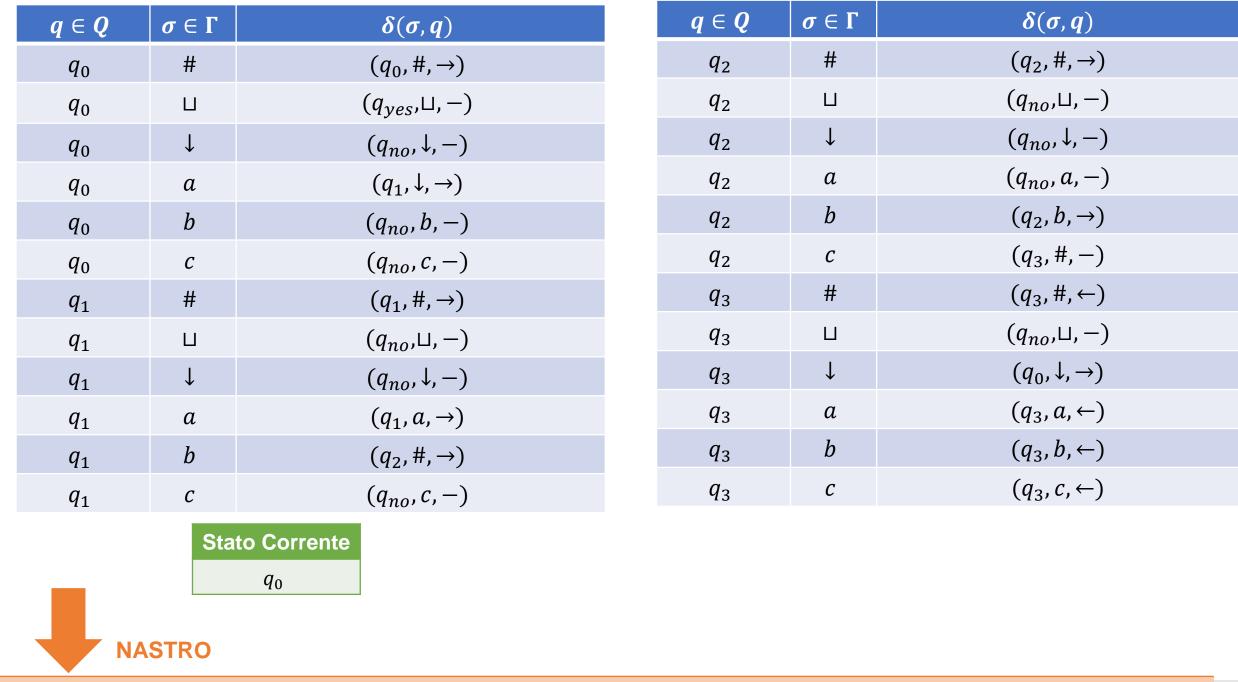
$q \in Q$	$\sigma \in \Gamma$	$\boldsymbol{\delta}(\pmb{\sigma},\pmb{q})$
$q_0$	#	$(q_0, \#, \rightarrow)$
$q_0$	Ц	$(q_{yes}$ , $\sqcup$ , $-)$
$q_0$	$\downarrow$	$(q_{no},\downarrow,-)$
$q_0$	а	$(q_1,\downarrow,\rightarrow)$
$q_0$	b	$(q_{no},b,-)$
$q_0$	С	$(q_{no},c,-)$
$q_1$	#	$(q_1, \#, \rightarrow)$
$q_1$	Ц	$(q_{no},\sqcup,-)$
$q_1$	1	$(q_{no},\downarrow,-)$
$q_1$	a	$(q_1, a, \rightarrow)$
$q_1$	b	$(q_2, \#, \rightarrow)$
$q_1$	С	$(q_{no},c,-)$

$q \in Q$	$\sigma \in \Gamma$	$\delta(\sigma,q)$
$q_2$	#	$(q_2, \#, \rightarrow)$
$q_2$	Ц	$(q_{no},\sqcup,-)$
$q_2$	1	$(q_{no},\downarrow,-)$
$q_2$	а	$(q_{no},a,-)$
$q_2$	b	$(q_2, b, \rightarrow)$
$q_2$	С	$(q_3, \#, -)$
$q_3$	#	$(q_3,\#,\leftarrow)$
$q_3$	Ц	$(q_{no},\sqcup,-)$
$q_3$	$\downarrow$	$(q_0,\downarrow,\rightarrow)$
$q_3$	а	$(q_3, a, \leftarrow)$
$q_3$	b	$(q_3, b, \leftarrow)$
$q_3$	С	$(q_3, c, \leftarrow)$

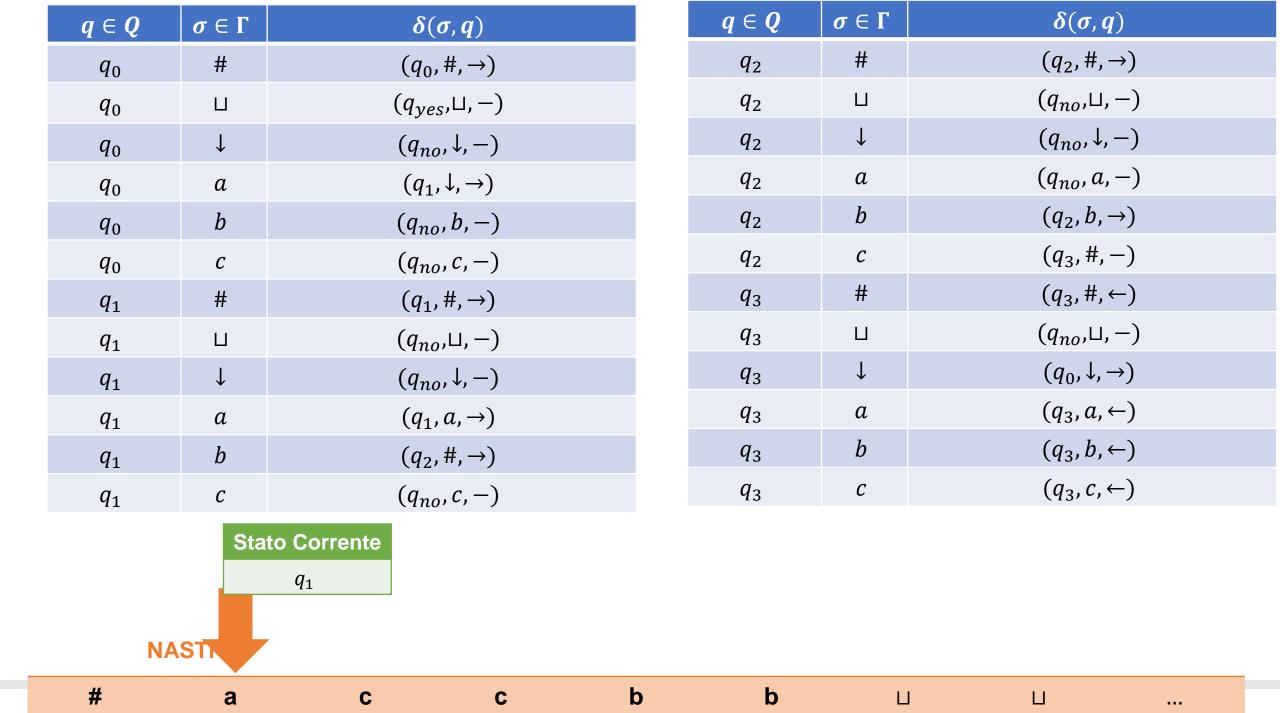
#### **NASTRO**

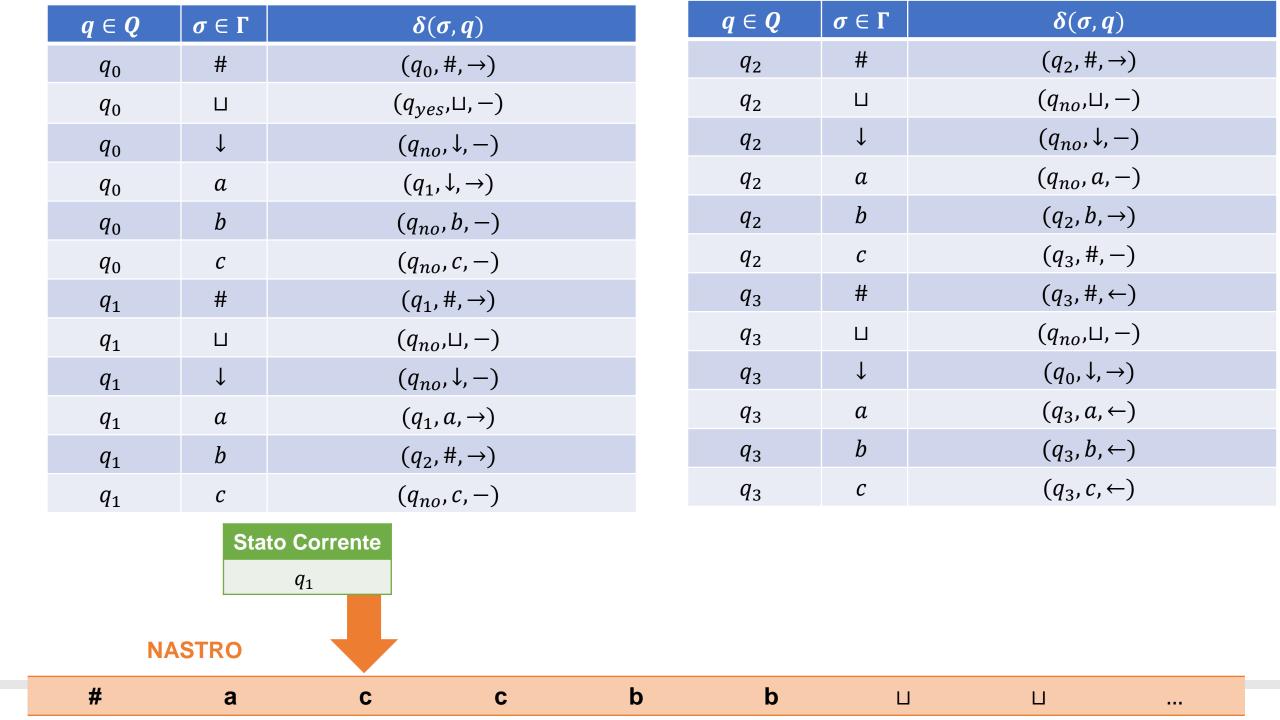
a a c c b b u u ...

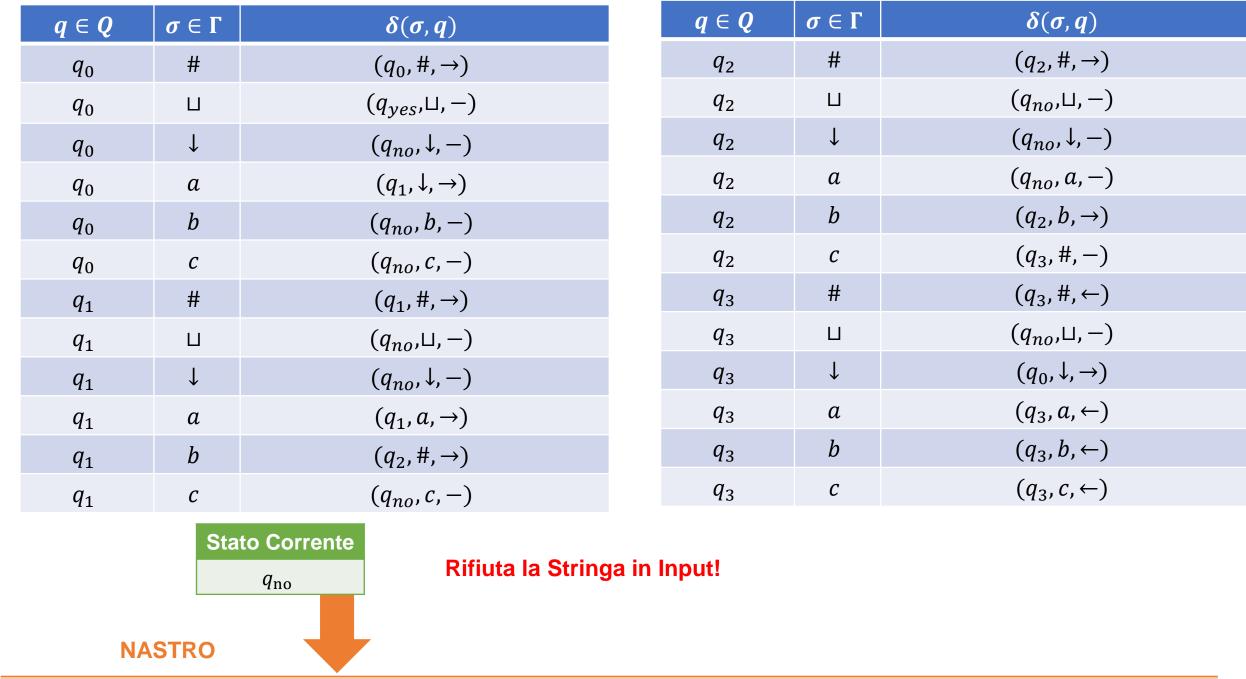




# a c c b b 🗆 ...







# a c c b b 🗆 ...

# Descrizione Semi-Formale delle Macchinen di Turing

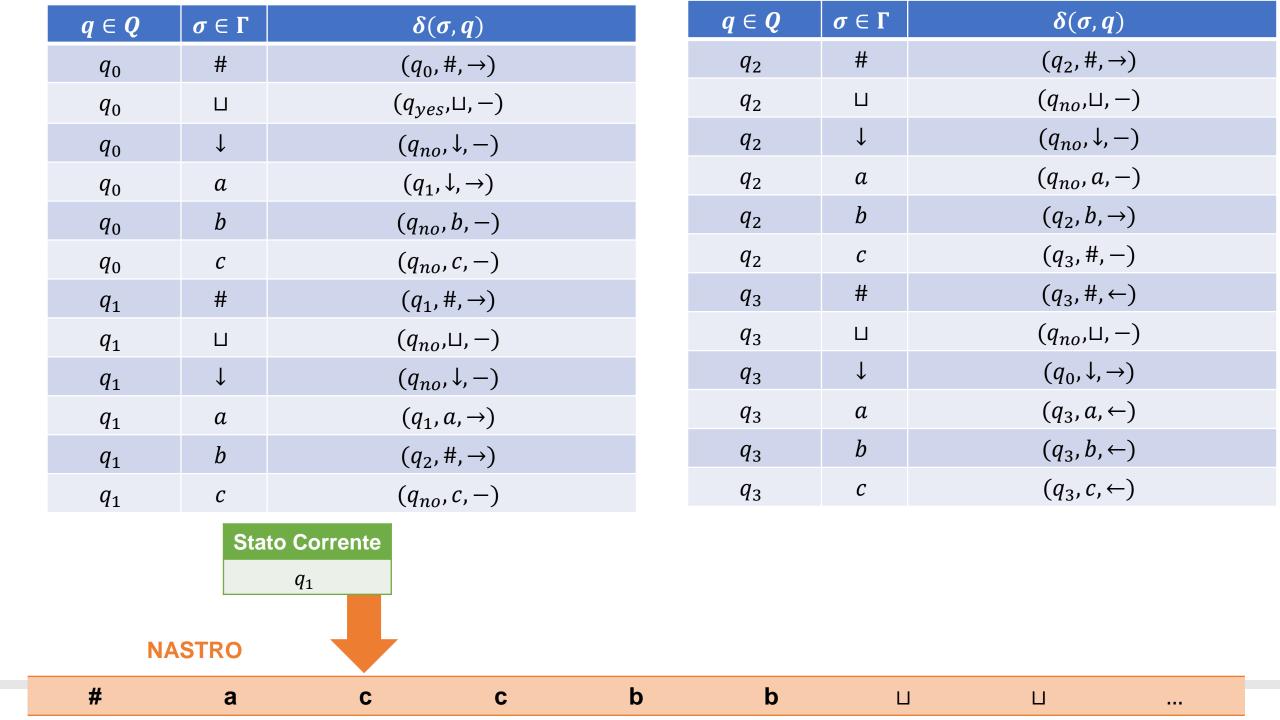
#### Macchina di Turing – Descrizione Semi Formale

- La definizione di una macchina di Turing richiede tantissimi dettagli di basso livello
  - Movimento delle testine, stati interni ecc.
- Ovviamente, non vogliamo definire tali dettagli ogni volta che definiamo un algoritmo per riconoscere un determinato linguaggio
  - Equivalentemente, per risolvere un problema
- Spesso, utilizziamo una descrizione semi-formali delle macchine
  - Basata sul linguaggio naturale che ne spiega le funzionalità
  - In maniera simile allo pseudo-codice di un algoritmo
- Nel costruire tale descrizione, dobbiamo prestare attenzione ad utilizzare solo le funzionalità ammesse da un Macchina di Turing
  - Così che la conversione sia ragionevolmente semplice

## Macchine di Turing – Esempio

#### Algoritmo per riconoscere $L = \{a^m b^m c^m | m \ge 0\}$

- 1. Leggi il prossimo carattere x della stringa dell'input
- 2. Se x = # Allora passa al prossimo carattere
- 3. Se  $x = \sqcup$  (cella vuota) Allora accetta la stringa corrente
- 4. Se x = b Allora rifiuta la stringa corrente
- 5. Se x = c Allora rifiuta la stringa corrente
- **6.** Se x = a Allora
  - 1. Sovrascrivi tale occorrenza di  $\alpha$  con il carattere  $\downarrow$
  - 2. Scorri il nastro fino alla prossima occorrenza carattere b
    - 1. Se incontri un carattere  $y \in \{c, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti sovrascrivi tale occorrenza di b con #
  - 3. Scorri il nastro fino al prossimo *c* 
    - **1.** Allora incontri un carattere  $y \in \{a, b, \sqcup\}$  Allora rifiuta la stringa corrente
    - 2. Altrimenti cambia tale occorrenza di c con #
  - **4.** Torna all'ultima occorrenza del carattere ↓



## Macchine di Turing – Esempio

- 1. Inizia dallo stato  $q_0$
- 2. Nello stato  $q_0$ 
  - 1. Se il carattere corrente è # allora muovi la testina a destra
  - 2. Se il carattere corrente è  $\sqcup$  (cella vuota) allora passa allo stato  $q_{yes}$
  - 3. Se il carattere corrente è b allora passa allo stato  $q_{no}$
  - 4. Se il carattere corrente è c allora passa allo stato  $q_{no}$
  - 5. Se il carattere corrente è  $\downarrow$  allora passa allo stato  $q_{no}$
  - 6. Se il carattere corrente è a Allora
    - 1. Sovrascrivi il carattere corrente con ↓
    - 2. Passa allo stato  $q_1$
- 3. Nello stato  $q_1$ 
  - 1. Se il carattere corrente è # allora muovi la testina a destra
  - 2. Se il carattere corrente è  $\sqcup$  (cella vuota) allora passa allo stato  $q_{no}$
  - 3. Se il carattere corrente è a allora passa allo stato  $q_{no}$
  - 4. Se il carattere corrente è c allora passa allo stato  $q_{no}$
  - 5. Se il carattere corrente è  $\downarrow$  allora passa allo stato  $q_{no}$
  - 6. Se il carattere corrente è b Allora
    - 1. Sovrascrivi il carattere corrente con #
    - 2. Passa allo stato  $q_2$

# Macchine di Turing – Esempio

#### 1. Nello stato $q_2$

- 1. Se il carattere corrente è # allora muovi la testina a destra
- 2. Se il carattere corrente è  $\sqcup$  (cella vuota) allora passa allo stato  $q_{no}$
- 3. Se il carattere corrente è a allora passa allo stato  $q_{no}$
- 4. Se il carattere corrente è b allora passa allo stato  $q_{no}$
- 5. Se il carattere corrente è  $\downarrow$  allora passa allo stato  $q_{no}$
- 6. Se il carattere corrente è c Allora
  - 1. Sovrascrivi il carattere corrente con #
  - 2. Passa allo stato  $q_3$

#### 2. Nello stato $q_3$

- 1. Se il carattere corrente è # allora muovi la testina a sinistra
- 2. Se il carattere corrente è  $\sqcup$  (cella vuota) allora passa allo stato  $q_{no}$
- 3. Se il carattere corrente è  $\alpha$  allora muovi la testina a sinistra
- 4. Se il carattere corrente è b allora muovi la testina a sinistra
- 5. Se il carattere corrente è c allora muovi la testina a sinistra
- 6. Se il carattere corrente è ↓ Allora
  - 1. Muovi la testina a destra
  - 2. Passa allo stato  $q_0$

## Macchina di Turing – Considerazioni Finali

- 1. La prima descrizione è la meno formale
  - Definisce solo i passi generali dell'algoritmo
  - Sappiamo però che possiamo implementarli con una Macchina di Turing
- 2. La seconda descrizione è la più formale
  - Definisce tutte le transizioni in termini di stati e movimenti della testina
  - Possiamo dimostrare formalmente che riconosce il linguaggio che vogliamo
- 3. La terza descrizione è semi-formale
  - Definisce le transizioni in linguaggio naturale
  - Possiamo usarla nelle nostre dimostrazioni (con attenzione)

## Macchina di Turing – Considerazioni Finali

- Quale descrizione utilizzare dipende dal contesto
  - E tendo presente che le Macchine di Turing non sono un linguaggio di programmazione
- 1. Per fornire una intuizione delle nostre idee algoritmiche, usiamo la prima descrizione
  - Basta pensarci un po' per convincersi che possiamo implementare tutto con una Macchine di Turing anche se non stiamo fornendo i dettagli necessari
- 2. Per fornire una prova di un enunciato utilizziamo la terza descrizione
  - Possiamo utilizzarla nelle dimostrazioni perché i dettagli sono (quasi) tutti li
  - Una dimostrazione non è un programma che dobbiamo compilare ed eseguire
  - C'è sempre un essere umano di mezzo che può capire cosa stiamo dicendo
- 3. La seconda descrizione non è quasi mai utilizzata in letteratura
  - A meno di voler dimostrare proprietà delle macchine stesse oppure ...
  - Se vogliamo fornire in input una MdT ad un programma che abbiamo scritto