

Tutorato di Automi e Linguaggi Formali

Homework 1: DFA, NFA ed ϵ -NFA, conversioni ed operazioni su linguaggi

Gabriel Rovesti

Corso di Laurea in Informatica - Università degli Studi di Padova

Tutorato 1 - 10-03-2025

1 Progettazione di DFA

Esercizio 1. Progettare un DFA sull'alfabeto $\Sigma = \{0, 1\}$ che riconosca il linguaggio

$$L_1 = \{w \in \{0, 1\}^* \mid w \text{ termina con } 10\}.$$

- a) Disegnare il diagramma degli stati.
- b) Fornire la tabella di transizione completa.

Esercizio 2. Progettare un DFA sull'alfabeto $\Sigma = \{0, 1\}$ che riconosca il linguaggio

$$L_2 = \{w \in \{0, 1\}^* \mid w \text{ contiene un numero pari di } 1 \text{ e un numero dispari di } 0\}.$$

- a) Disegnare il diagramma degli stati.
- b) Fornire la tabella di transizione completa.

Esercizio 3. Progettare un DFA sull'alfabeto $\Sigma = \{0, 1\}$ che riconosca il linguaggio

$$L_3 = \{w \in \{0, 1\}^* \mid \text{ogni occorrenza di } 0 \text{ è immediatamente seguita da almeno due } 1\}.$$

- a) Disegnare il diagramma degli stati.
- b) Fornire la tabella di transizione completa.

2 Progettazione di NFA

Esercizio 4. Progettare un NFA sull'alfabeto $\Sigma = \{a, b\}$ che riconosca il linguaggio

$$L_4 = \{w \in \{a, b\}^* \mid w \text{ contiene la sottostringa } aba\}.$$

- Disegnare il diagramma degli stati.
- Fornire la tabella di transizione completa.

Esercizio 5. Progettare un NFA sull'alfabeto $\Sigma = \{0, 1\}$ che riconosca il linguaggio

$$L_5 = \{w \in \{0, 1\}^* \mid w \text{ inizia con 1 oppure termina con 0}\}.$$

- Disegnare il diagramma degli stati.
- Fornire la tabella di transizione completa.

3 Conversione da NFA a DFA

Esercizio 6. Si consideri il seguente NFA N sull'alfabeto $\Sigma = \{a, b\}$:

| | a | b |
|-------------------|----------------|-----------|
| $\rightarrow q_0$ | $\{q_0, q_1\}$ | $\{q_0\}$ |
| q_1 | \emptyset | $\{q_2\}$ |
| $*q_2$ | $\{q_2\}$ | $\{q_1\}$ |

- Applicare la costruzione per sottoinsiemi per ottenere il DFA equivalente D .
- Disegnare il diagramma degli stati del DFA ottenuto.
- Determinare il linguaggio riconosciuto dall'automa.

4 ϵ -NFA e ϵ -chiusure

Esercizio 7. Dato il seguente ϵ -NFA sull'alfabeto $\Sigma = \{a, b\}$:

| | a | b | ϵ |
|-------------------|-------------|-------------|----------------|
| $\rightarrow q_0$ | \emptyset | \emptyset | $\{q_1, q_3\}$ |
| q_1 | $\{q_2\}$ | \emptyset | \emptyset |
| q_2 | \emptyset | $\{q_4\}$ | $\{q_3\}$ |
| q_3 | $\{q_5\}$ | \emptyset | \emptyset |
| q_4 | \emptyset | $\{q_5\}$ | \emptyset |
| $*q_5$ | \emptyset | \emptyset | \emptyset |

- Calcolare l' ϵ -chiusura di ciascuno stato: $\text{ECLOSE}(q_0)$, $\text{ECLOSE}(q_1)$, $\text{ECLOSE}(q_2)$, $\text{ECLOSE}(q_3)$, $\text{ECLOSE}(q_4)$, $\text{ECLOSE}(q_5)$.
- Convertire l' ϵ -NFA in un NFA equivalente senza ϵ -transizioni, fornendo la tabella di transizione completa.

5 Operazioni su Linguaggi e Automi

Esercizio 8. Siano $L_1 = \{w \in \{0, 1\}^* \mid w \text{ inizia con } 0\}$ e $L_2 = \{w \in \{0, 1\}^* \mid w \text{ termina con } 1\}$.

- a) Progettare un DFA che riconosca L_1 .
- b) Progettare un DFA che riconosca L_2 .
- c) Utilizzando le operazioni sui linguaggi regolari, costruire un NFA che riconosca il linguaggio $L_1 \cup L_2$.
- d) Convertire il NFA ottenuto in un DFA equivalente mediante la costruzione per sottoinsiemi.

Esercizio 9. Siano $L_1 = \{w \in \{a, b\}^* \mid \text{ogni } a \text{ è seguita da almeno una } b\}$ e $L_2 = \{w \in \{a, b\}^* \mid w \text{ contiene la sottostringa } ab\}$.

- a) Progettare un DFA per L_1 e un NFA per L_2 .
- b) Costruire un NFA che riconosca $L_1 \cap L_2$.
- c) Qual è l'interpretazione di questo linguaggio in linguaggio naturale?

Esercizio 10. Sia $L = \{w \in \{0, 1\}^* \mid w \text{ contiene un numero pari di } 0\}$.

- a) Progettare un DFA A che riconosca L .
- b) Costruire un DFA che riconosca il complemento di L , ovvero $\bar{L} = \{w \in \{0, 1\}^* \mid w \text{ contiene un numero dispari di } 0\}$.
- c) Costruire un NFA che riconosca $L^* = \{w_1 w_2 \dots w_k \mid k \geq 0 \text{ e } w_i \in L \text{ per ogni } i\}$.