

# Automi e Linguaggi Formali

Università degli Studi di Padova

Marzo 2024

---

## Homework 2 - Espressioni regolari, Equivalenze con automi, Conversioni

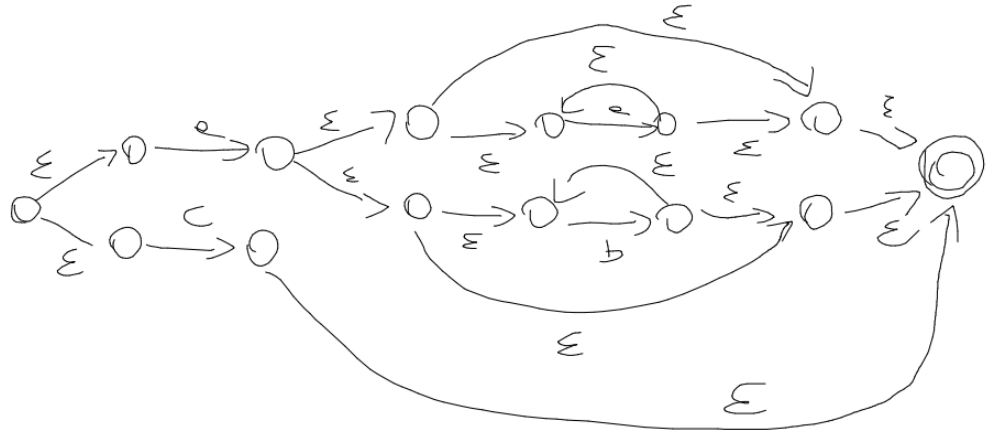
Gabriel Rovesti

1. Fornite espressioni regolari per i seguenti linguaggi:

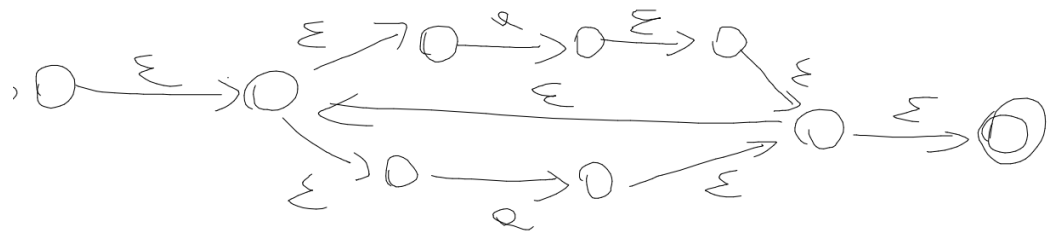
- $L_1 = \{w \mid w \text{ ha un numero pari di "a" e una o due "b"}\}$   
–  $(aa)^*((b|bb))$
- $L_1 = \{w \mid w \text{ stringhe binarie il cui quarto simbolo è uno zero.}\}$   
–  $(0+1)(0+1)(0+1)0(0+1)$
- $L_1 = \{w \mid w \text{ ha un numero pari di "a" e un numero dispari di "b"}\}$   
–  $(b+a(aa+bb)^*(ab+ba))((ab+ba)(aa+bb)^*(ab+ba)+(aa+bb))^*$
- $L_1 = \{w \mid w \text{ inizia con 0, ha una lunghezza dispari oppure inizia con 1 e ha lunghezza pari}\}$   
–  $(0(00)^*1|(1(11)^*0)^*)+$
- $L_1 = \{w \mid w \text{ accetta tutte le stringhe su } \Sigma = a, b \text{ che contengono esattamente 2 oppure 3 lettere 'b'}\}$   
–  $a^*ba^*ba^* + a^*ba^*ba^*ba^*$
- $L_1 = \{w \mid w \text{ accetta tutte le stringhe con numero di 0 multiplo di 5}\}$   
–  $(1+(01^*01^*01^*01^*0))^*$

2. Convertite le seguenti espressioni regolari in automi a stati finiti:

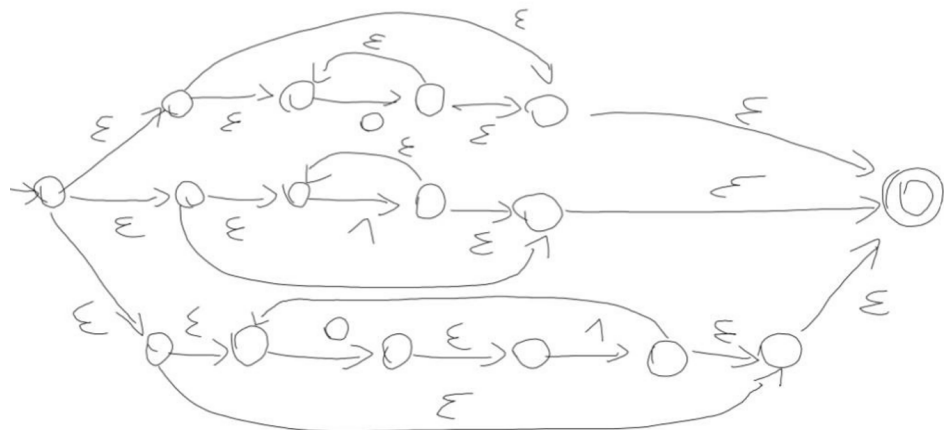
- $a(a^* + b^*) + c$



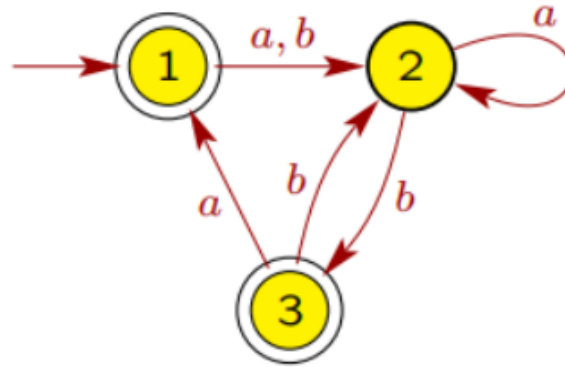
- $(ab + a)^*$



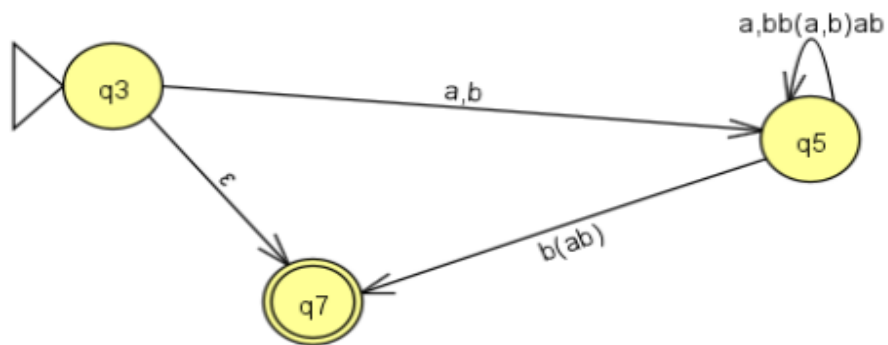
- $0^* + 1^* + (01)^*$



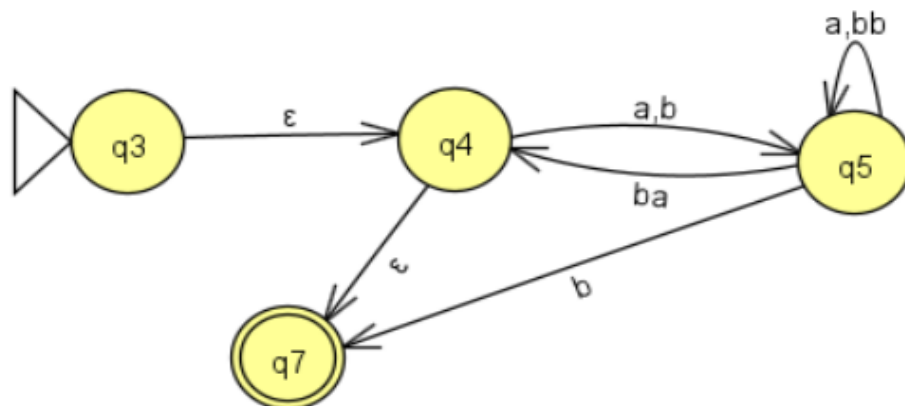
3. Convertite i seguenti automi DFA in un'espressione regolare usando l'algoritmo di eliminazione degli stati:



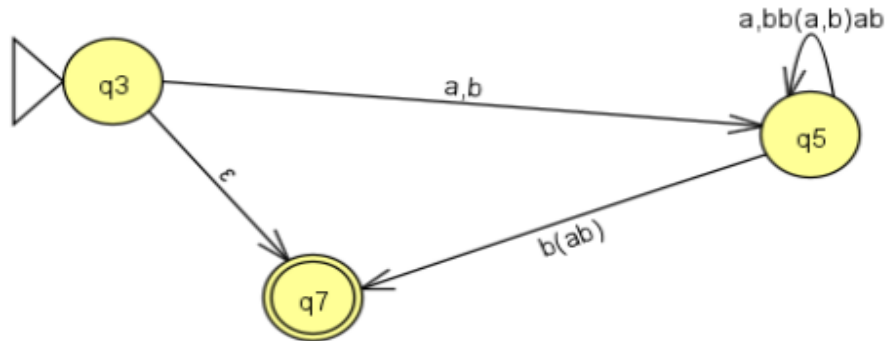
Primo automa



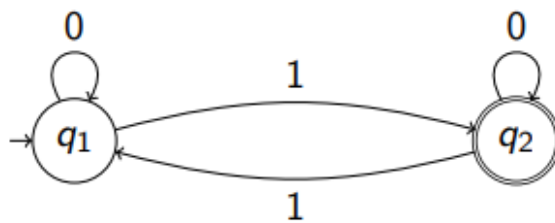
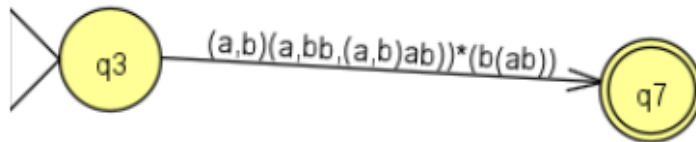
Eliminiamo q6



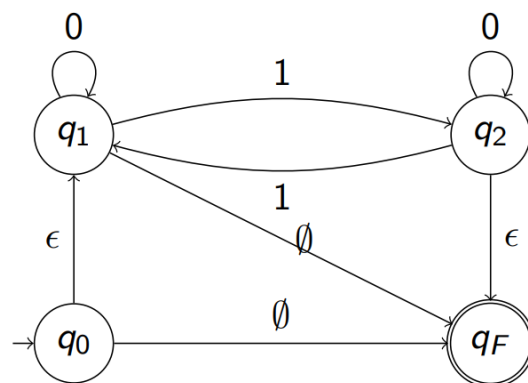
Eliminiamo poi q4



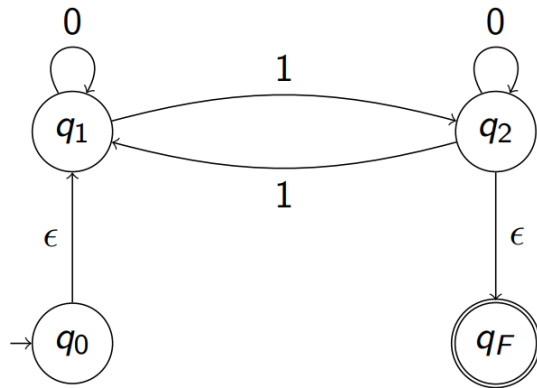
Eliminiamo quindi q5 e la regex finale è:  $(a,b)(a,bb,(a,b)ab))^*(b(ab))$



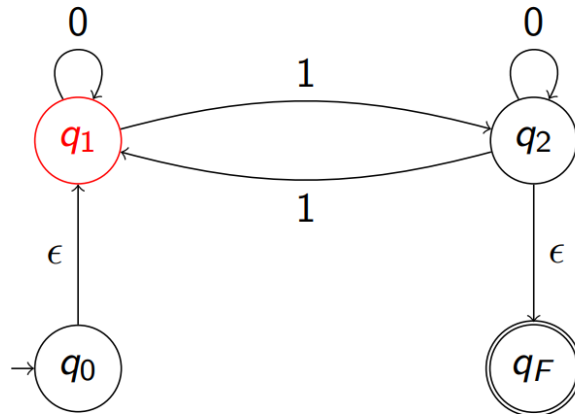
Secondo automa



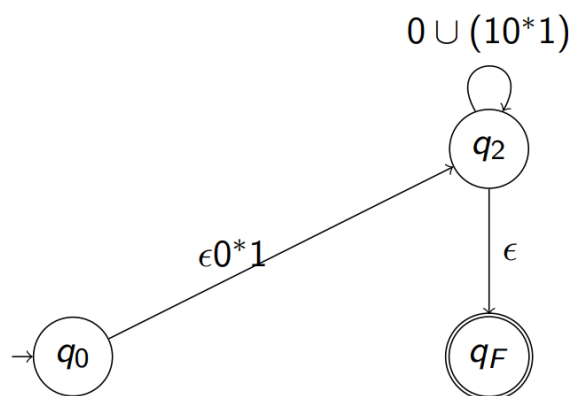
Primo passaggio



Secondo passaggio



Terzo passaggio



Quarto passaggio

