

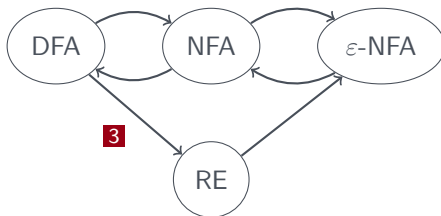
Automi e Linguaggi Formali

a.a. 2016/2017

LT in Informatica
20 Marzo 2017

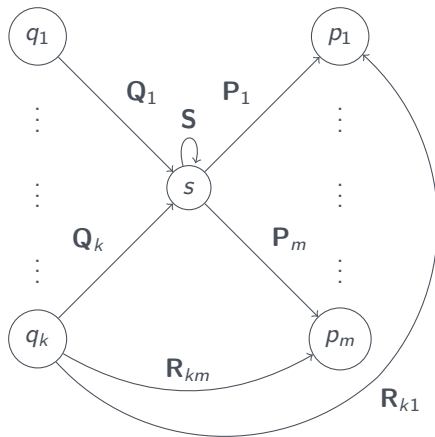


UNIVERSITÀ
DEGLI STUDI
DI PADOVA



- A** Sappiamo già che DFA, NFA, e ε -NFA sono tutti equivalenti.
- B** Per ogni espressione regolare R esiste un ε -NFA A , tale che $L(A) = L(R)$
- C** Per ogni FA A possiamo costruire un'espressione regolare R , tale che $L(R) = L(A)$

- La procedura che vedremo è in grado di convertire un **qualsiasi automa** (DFA, NFA, ε -NFA) in una **espressione regolare** equivalente
- Si procede per **eliminazione di stati**
- Quando uno stato q viene eliminato, i **cammini** che passano per q scompaiono
- si aggiungono nuove **transizioni etichettate con espressioni regolari** che rappresentano i cammini eliminati
- alla fine otteniamo un'espressione regolare che rappresenta **tutti i cammini** dallo stato iniziale ad uno stato finale
⇒ cioè il **linguaggio riconosciuto dall'automata**



- Lo stato da eliminare può avere un **ciclo**
- q_1, \dots, q_k sono i **predecessori**
- p_1, \dots, p_m sono i **successori**
- ci possono essere **transizioni dirette** tra i predecessori ed i successori

q_1

\vdots

\vdots

\vdots

q_k

p_1

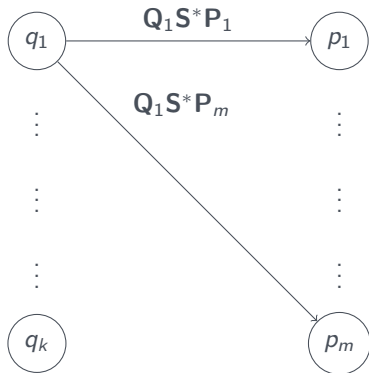
\vdots

\vdots

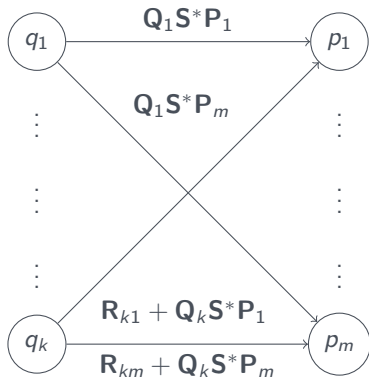
\vdots

p_m

- Dobbiamo ricreare la transizione per ogni coppia predecessore-successore q_i, p_j



- Dobbiamo **ricreare la transizione** per ogni **coppia predecessore-successore** q_i, p_j
- Se non c'è la transizione diretta, l'etichetta è **$Q_i S^* P_j$**



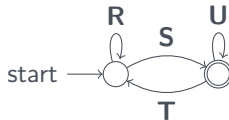
- Dobbiamo **ricreare la transizione** per ogni **coppia predecessore-successore** q_i, p_j
- Se non c'è la transizione diretta, l'etichetta è $Q_i S^* P_j$
- Se c'è la transizione diretta, l'etichetta è $R_{ij} + Q_i S^* P_j$

- Per ogni stato finale $q \in F$:
 - 1 elimina tutti gli stati tranne lo stato iniziale q_0 e q

■ Per ogni stato finale $q \in F$:

1 elimina tutti gli stati **tranne lo stato iniziale q_0 e q**


2 se $q \neq q_0$ l'automa finale è



che è equivalente a $(R + SU^*T)^*SU^*$

■ Per ogni stato finale $q \in F$:

1 elimina tutti gli stati **tranne lo stato iniziale q_0 e q**

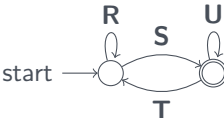
2 se $q \neq q_0$ l'automa finale è 

che è equivalente a $(R + SU^*T)^*SU^*$


3 se $q = q_0$ l'automa finale è 
che è equivalente a R^*

- Per ogni stato finale $q \in F$:

1 elimina tutti gli stati **tranne lo stato iniziale q_0 e q**

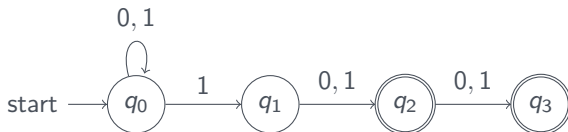
2 se $q \neq q_0$ l'automa finale è 

che è equivalente a $(R + SU^*T)^*SU^*$

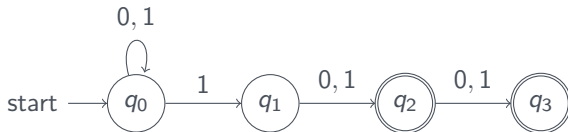
3 se $q = q_0$ l'automa finale è 
che è equivalente a R^*

- L'espressione regolare desiderata è l'**unione** (+) di tutte le espressioni ottenute dalle regole 2 e 3

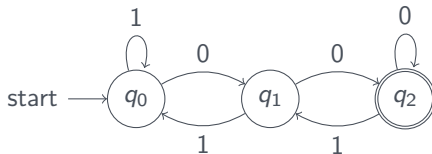
- 1** Costruiamo l'espressione regolare equivalente al seguente NFA:



- 1** Costruiamo l'espressione regolare equivalente al seguente NFA:



- 2** Costruiamo l'espressione regolare equivalente al seguente NFA:



- 3** Costruiamo l'espressione regolare equivalente al seguente NFA:

