LEZIONE 04/03/25

Definizione più formale: se r ed s sono espressioni regolari:

- $L(\emptyset) = \emptyset$
- $L(\varepsilon) = \{\varepsilon\}$
- $L(a) = \{a\}, \forall a \in \Sigma$
- $L(r+s) = L(r) \cup L(s)$
- $L(rs) = L(r)L(s)$
- $L(r^*) = L(r)^*$

$$L(\varepsilon) = L(\emptyset^*) = L(\emptyset)^* = \{\}^* = \{\varepsilon\}$$

ESEMPI: Sia $\Sigma = \{0,1\}$. Il linguaggio associato all'espressione regolare $r = 1(0+1)^*00$ e'

$$L(r) = L(1)(L(0) \cup L(1))^* L(0)L(0) =$$

$$= \{1\}(\{0\} \cup \{1\})^* \{0\}\{0\} =$$

$$= \{1\}(\{0\} \cup \{1\})^* \{00\} =$$

$$= \{1\}(\{0,1\})^* \{00\}$$

$$= \{1x00 \mid x \in \Sigma^*\}$$

ovvero e' insieme di stringhe binarie rappresentanti un multiplo di 4.

ESERCIZIO 4.8:

$\Sigma = \{0,1\}$

1) stringhe aventi 3 zeri consecutivi:

$$r1 = (0+1)^* 000 (0+1)^* = \{x000y \mid x, y \in \Sigma^*\}$$

indica che prima e dopo può esserci qualunque cosa

2) stringhe tali che l'ultimo simbolo e' zero:

$$r2 = (0+1)^* 0 = \{x0 \mid x \in \Sigma^*\}$$

3) il terzultimo simbolo e' zero:

$$r3 = (0+1)^*0(0+1)(0+1) = \{x0ab \mid x \in \Sigma^*, a,b \in \Sigma^*\}$$

4) stringhe che se implementate come numero binario sono divisibili per 2

$$r4 = (0+1)^*0 = \{x0 \mid x \in \Sigma^*$$

5) stringhe che se implementate come numero binario sono divisibili per 4

$$r5 = (0+1)^*00 = \{x00 \mid x \in \Sigma^*\}$$

PLUS) stringhe aventi esattamente 3 zeri

$$r = 1^*01^*0\,1^*01^*$$
$$= (1^*)0(1^*)0(1^*)0(1^*)$$

PLUS) stringhe aventi almeno 3 zeri

$$r = (0+1)^*0(0+1)^*0(0+1)^*0(0+1)^*$$

es. <u>CODICE FISCALE</u>

$$[A-Z] = A+B+C+D+\ldots+Y+Z$$
$$[0-9] = 1+2+3\ldots+8+9$$

$$r\_cf = \underbrace{[A-Z][A-Z][A-Z]}_{\text{cognome}}\underbrace{[A-Z][A-Z][A-Z]}_{\text{nome}}\underbrace{[0-9][0-9]}_{\text{anno di nascita}}\underbrace{[A-Z]}_{\text{mese}}$$

$$\underbrace{[0-9]}_{\text{giorno}}\underbrace{[0-9]}_{\text{sesso}}\underbrace{[A-Z][0-9][0-9][0-9]}_{\text{codice catastale}}\underbrace{[A-Z]}_{\text{carattere di controllo}}$$

EQUIVALENZA TRA DFA ed ESPRESSIONI REGOLARI

TEOREMA

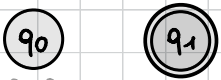
Sia r una espressione regolare. Allora esiste un ε-NFA M tale che $L(M) = L(r)$.

DIMOSTRAZIONE: Costruiamo un ε-NFA, con un unico stato finale, per induzione sulla struttura dell'espressione regolare r.

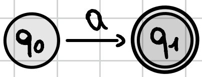
BASE, 3 casi:

1) l'automa 

riconosce il linguaggio $\{\varepsilon\}$;

2) l'automa 

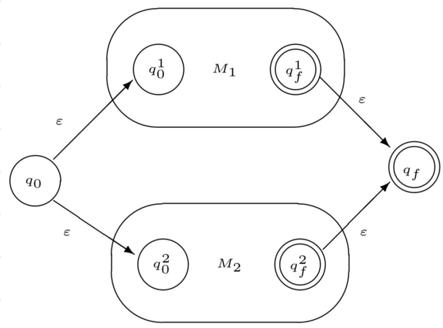
riconosce il linguaggio $\emptyset$;
(rifiuta qualunque cosa venga letta) (anche se non raggiungeremo mai q_1, abbiamo fatto la premessa di avere sempre uno stato accettante)

3) l'automa 

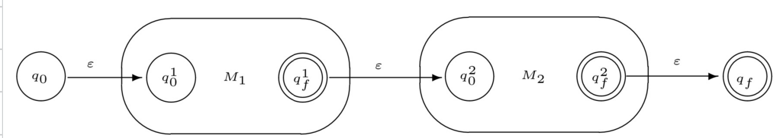
riconosce il linguaggio $\{a\}$.

PASSO, 3 casi:

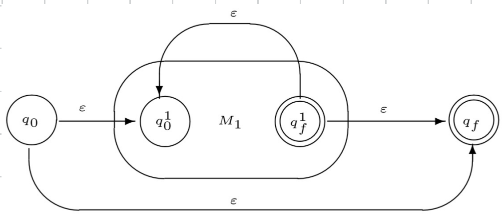
1) $r = r_1 + r_2$. Per $i = 1, 2$, sia M_i, con stato iniziale q_0^i e stato finale q_f^i l'automa che riconosce $L(r_i)$.
Il seguente automa, con stato iniziale q_0 e stato finale q_f riconosce il linguaggio $L(r) = L(r_1) \cup L(r_2)$:



2) $r = r_1 r_2$. Per $i = 1, 2$, sia M_i con stato iniziale q_0^i e stato finale q_f^i l'automa che riconosce $L(r_i)$. L'esistenza di tali automi e' assicurata dall'ipotesi induttiva. Il seguente automa, con stato iniziale q_0 e stato finale q_f riconosce il linguaggio $L(r_1)L(r_2)$:



3) $r = r_1^*$. Sia M_1, con stato iniziale q_0^1 e stato finale q_f^1 l'automa che riconosce $L(r_1)$. L'esistenza di tale automa è assicurata dall'ipotesi induttiva. Il seguente automa, con stato iniziale q_0 e stato finale q_f riconosce $L(r)=(L(r_1))^*$



<u>APPLICAZIONE</u> :

$r = a + bc$ $\qquad\qquad$ $\Sigma = \{a, b, c\}$

$= (a) + (bc)$

e per b e c

serve un automa per a