TUTORIAT LFA 2: OPERAȚII CU AUTOMATE. TRANSFORMĂRI ÎNTRE AUTOMATE FINITE

RADU COSTACHE, MARIA PREDA

1. Breviar Teoretic

1.1. Închiderea la operaţii.

Teoremă 1.1. *Pentru orice NFA, există un DFA echivalent.*

Teoremă 1.2. *Pentru orice λ-NFA, există un NFA echivalent.*

Observaţie 1.3. *Pentru orice λ-NFA, există un DFA echivalent.*

Definiţie 1.4. *Se numeşte limbaj regulat, un limbaj pentru care se poate construi un automat finit. Vom nota familia limbajelor regulate cu* REG.

Observaţie 1.5. *Familia* REG *este închisa la următoarele operaţii:*

- *Reuniune*
- *Intersecţie*
- *Diferenţă*
- *Complementare*
- *Concatenare*
- *Stelare*
- *Plus*

1.2. Construcţia automatelor pentru operaţii unare.

Fie $A = (Q, \Sigma, q_0, F, \delta)$ un automat finit. . Realizăm următoarele operaţii:

- **Complement**: $\overline{A} = (Q, \Sigma, q_0, Q \setminus F, \delta)$, doar dacă A este **DFA complet definit**.
- **Stelare**: Din fiecare stare finală se adaugă o $\lambda-$tranziţie către starea iniţială şi $F' = F \cup \{q_0\}$
- **Plus**: Din fiecare stare finală se duce o $\lambda-$tranziţie spre starea iniţialăşi $F' = F$.

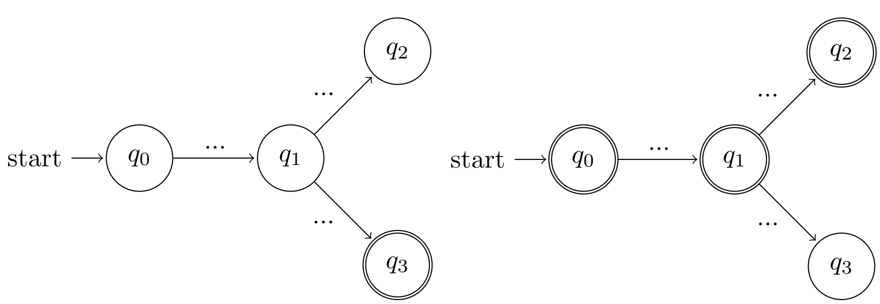


Figura 1. Complementare DFA complet definit

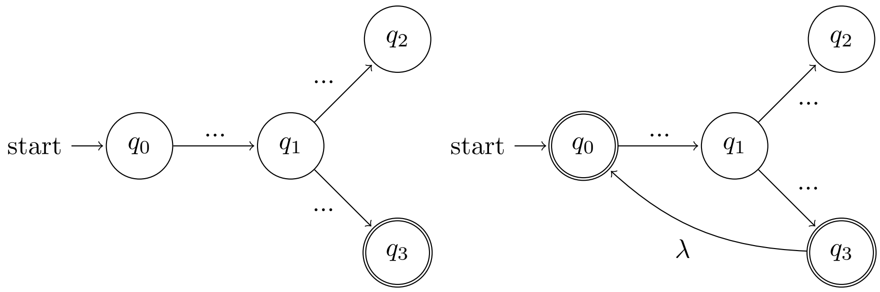


FIGURA 2. Stelare automat

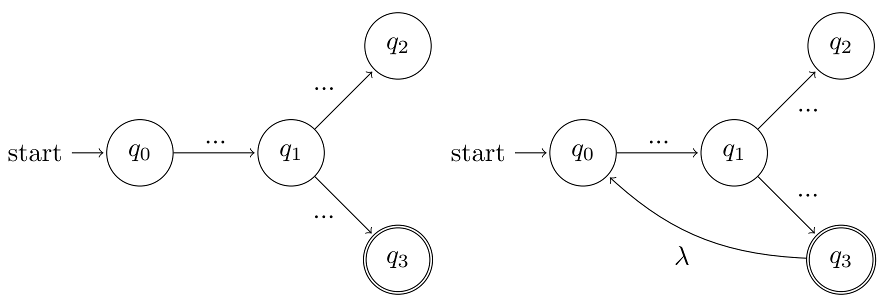
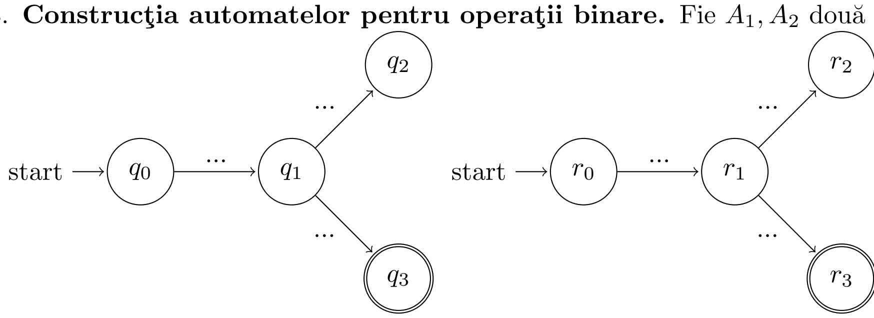


FIGURA 3. Operaţia Plus.

1.3. **Construcţia automatelor pentru operaţii binare.** Fie A_1, A_2 două automate finite.



Dacă este suficient să obţinem λ-NFA, putem proceda astfel, pentru operaţiile binare:

- **Reuniune**: Creăm o stare iniţialănouă (s_0). Ducem λ-tranziţii către ambele stări iniţiale.
- **Concatenare**: Adăugăm câte o λ-tranziţie din fiecare stare finală a primului automat spre starea iniţială a celui de-al doilea.

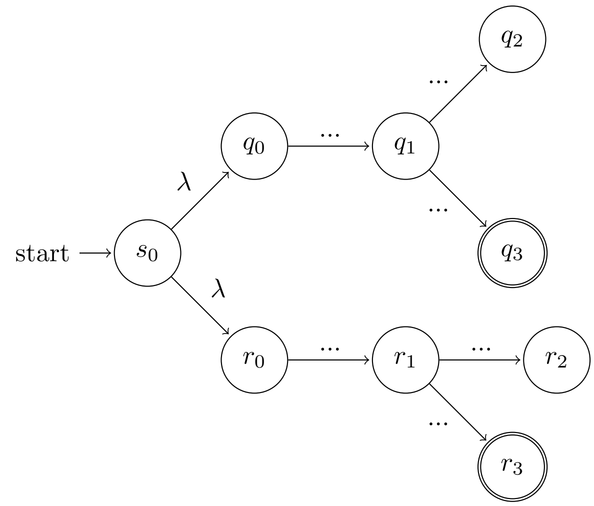


FIGURA 4. Reuninea limbajelor automatelor $L(A_1) \cup L(A_2)$.

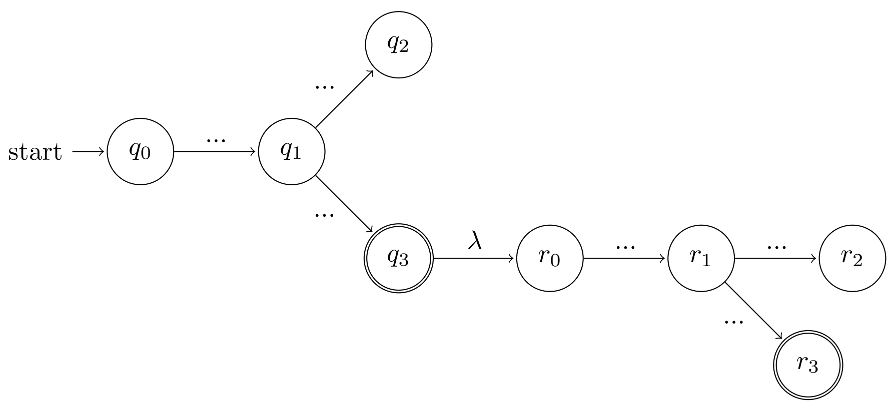


FIGURA 5. Concatenarea limbajelor automatelor $L(A_1) \cdot L(A_2)$.

1.4. **Construcţia automatelor pentru operaţii binare cu rezultatat DFA.** Fie

$$A_1 = (Q_1 = \{q_0, q_1, q_2, \ldots\}, \Sigma_1, q_0, F_1, \delta_1)$$
$$A_2 = (Q_2 = \{r_0, r_1, r_2, \ldots\}, \Sigma_2, r_0, F_2, \delta_2)$$

două **DFA-uri complet definite**. Vrem să construim automatul

$$A = (Q, \Sigma, (q_0, r_0), F, \delta)$$

pentru fiecare din operaţiile:

- **Reuniune**:
  - Stările: $Q = Q1 \times Q_2$
  - Funcţia de tranziţie: $\delta\left(\left(q_i, r_j\right), x\right) = \left(\delta_1\left(q_i, x\right), \delta_2\left(r_j, x\right)\right), \forall \left(q_i, r_j\right) \in Q, \forall x \in \Sigma$
  - Starea iniţială: (q_0, r_0)
  - Stările finale: $F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$
- **Intersecţie**:
  - Stările: $Q = Q1 \times Q_2$
  - Funcţia de tranziţie: $\delta\left(\left(q_i, r_j\right), x\right) = \left(\delta_1\left(q_i, x\right), \delta_2\left(r_j, x\right)\right), \forall \left(q_i, r_j\right) \in Q, \forall x \in \Sigma$
  - Starea iniţială: (q_0, r_0)
  - Stările finale: $F = F_1 \times F_2$
- **Diferenţă**:

– Stările: $Q = Q1 \times Q_2$
– Funcţia de tranziţie: $\delta\left(\left(q_i, r_j\right), x\right) = \left(\delta_1\left(q_i, x\right), \delta_2\left(r_j, x\right)\right), \forall\left(q_i, r_j\right) \in Q, \forall x \in \Sigma$
– Starea iniţială: (q_0, r_0)
– Stările finale: $F = F_1 \times (Q_2 \setminus F_2)$

1.5. **Algoritmul de transformare a unui NFA în DFA.** Ştim ca principala diferenţă între un NFA şi un DFA constă in faptul că NFA-ul admite tranziţii, dintr-o anumită stare, cu un acelaşi simbol, către mai mult de o singură stare. Astfel, o idee intuitivă ar fi, ca de fiecare dată când putem ajunge dintr-o stare cu un simbol în mai multe stări, să transformăm toate aceste stări destinatie într-una singură.

Observaţie 1.6. *Stările noului DFA creat vor fi submulţimi ale mulţimii stărilor NFA-ului, starea iniţiala rămânând aceeaşi. Funcţia de tranziţie devine:*

$$\delta'(R, x) = \bigcup_{q \in R} \delta(q, x), \forall R \in P(Q), \forall x \in \Sigma$$

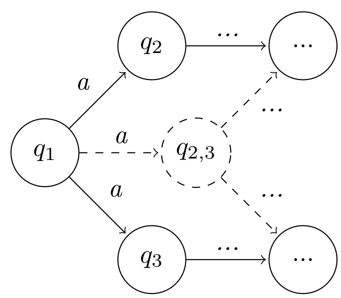
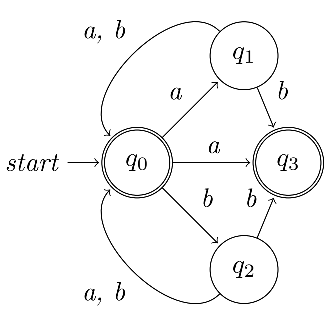


FIGURA 6. Stări DFA obţinute din unirea mai multor stări NFA

Observaţie 1.7. *Stările finale ale DFA-ului nou construit vor fi toate stările care, in componenţa lor, conţin cel puţin o stare care era înainte finala.*

$$F' = \{R \mid R \in Q', R \cap F \neq \emptyset\}$$

Exemplu 1.8. *Vom transforma urmatorul NFA intr-un DFA:*



Vom face intai tabelul pentru NFA, aratand pentru fiecare pereche (stare, simbol) in ce stari se poate ajunge.

Stare	a	b
q_0	$q_{13} = \{q_1, q_3\}$	q_2
q_1	q_0	$q_{03} = \{q_0, q_3\}$
q_2	q_0	$q_{03} = \{q_0, q_3\}$
q_3	$\emptyset$	$\emptyset$

Vom construi un tabel de tranzitii pentru noul DFA in felul urmator: pe prima linie punem starea initiala si starile in care se poate ajunge din ea cu fiecare simbol. Apoi, pe urmatoarea linie punem prima noua stare in care apare in tabel si tot asa, pana cand tuturor starilor in care se poate ajunge le corespunde o linie.

Stare	a	b
q_0	$q_{13} = \{q_1, q_3\}$	q_2
q_{13}	q_0	$q_{03} = \{q_0, q_3\}$
q_2	q_0	$q_{03} = \{q_0, q_3\}$
q_{03}	$q_{13} = \{q_1, q_3\}$	q_2

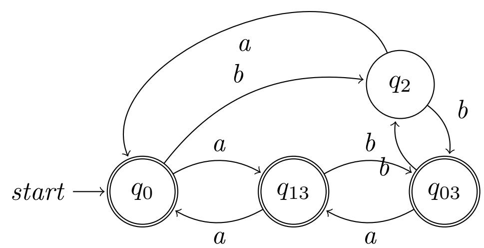


FIGURA 7. DFA-ul rezultat

1.6. **Algoritmul de transformare a unui λ-NFA în NFA.** Încercăm să facem este să eliminăm λ-tranziţiile. Deci, în automatul rămas, dintr-o stare in alta vom putea trece doar prin caractere din Σ diferite de λ.

Asa cum am vazut si in primul tutoritat, mulţimea de stări în care putem ajunge dintr-o stare cu un anumit caracter se numeşte λ-inchiderea stării şi se notează cu $\langle stare \rangle$.

Observaţie 1.9. *Orice stare face parte din propria λ-inchidere.*

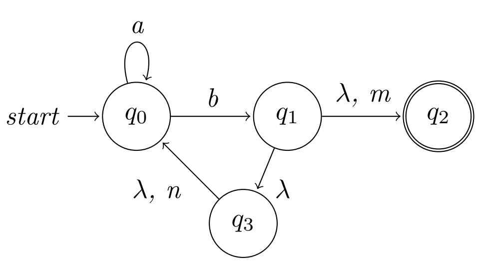
$$\langle q \rangle = \{r \mid r \in \hat{\delta}(q, \lambda^*)\}, \text{ unde } \hat{\delta} \text{ reprezinta compunerea mai multor tranzitii succesive}$$

Idee: Vom construi un nou automat astfel încât aflându-ne la o stare q, tranziţiile pentru un simbol c se duc înspre stările din λ-NFA în care puteam săajungem cu $\lambda^* c \lambda^*$.

Observaţie 1.10. *Starea iniţială rămâne aceeaşi.*

Observaţie 1.11. *Stările finale sunt acele stări care au în λ-închidere cel puţin o stare finală.*

Exemplu 1.12. *Fie următorul automat λ-NFA:*



Pas 1: Determinăm λ-închiderile:

$$\langle q_0 \rangle = \{q_0\}$$
$$\langle q_1 \rangle = \{q_1, q_2, q_3, q_0\}$$
$$\langle q_2 \rangle = \{q_2\}$$
$$\langle q_3 \rangle = \{q_3, q_0\}$$

Stările finale în NFA vor fi $F' = q_1, q_2$, deoarece $\langle q_1 \rangle \cap F = \{q_2\} \neq \emptyset$ şi $\langle q_2 \rangle \cap F = \{q_2\} \neq \emptyset$ (ambele îl conţin pe q_2 în λ-închidere, care este stare finală).

Pas 2: Realizăm tabelul de tranziţii:

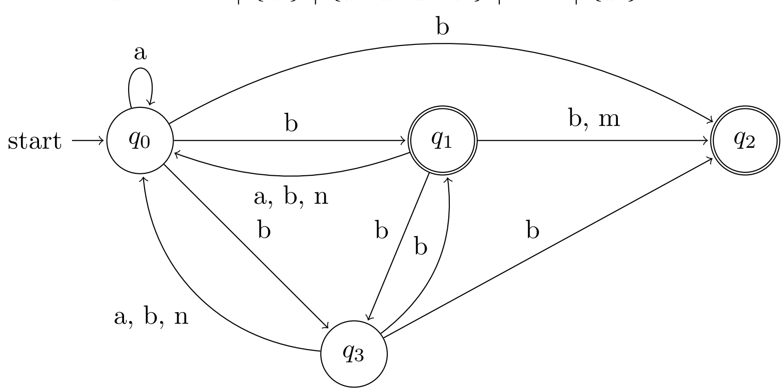
stare	a	b	m	n	λ	λ^*
q_0	$\{q_0\}$	$\{q_1\}$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_0\}$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\emptyset$	$\{q_2, q_3\}$	$\{q_1, q_2, q_3, q_0\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_2\}$
q_3	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_0\}$	$\{q_0\}$	$\{q_0, q_3\}$

Pas 3: Determinăm tranziţiile de forma $\lambda^* c \lambda^*$:

stare	$\lambda^* a \lambda^*$	$\lambda^* b \lambda^*$	$\lambda^* m \lambda^*$	$\lambda^* n \lambda^*$
q_0	$\{q_0\}$	$\{q_0, q_1, q_2, q_3\}$	$\emptyset$	$\emptyset$
q_1	$\{q_0\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_2\}$	$\{q_0\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
q_3	$\{q_0\}$	$\{q_0, q_1, q_2, q_3\}$	$\emptyset$	$\{q_0\}$

Pas 4: Realizăm tabelul de tranziţii pentru NFA, şi reprezentăm automatul:

stare	a	b	m	n
q_0 (starea iniţială)	$\{q_0\}$	$\{q_0, q_1, q_2, q_3\}$	$\emptyset$	$\emptyset$
$q_1 \in F$	$\{q_0\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_2\}$	$\{q_0\}$
$q_2 \in F$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
q_3	$\{q_0\}$	$\{q_0, q_1, q_2, q_3\}$	$\emptyset$	$\{q_0\}$



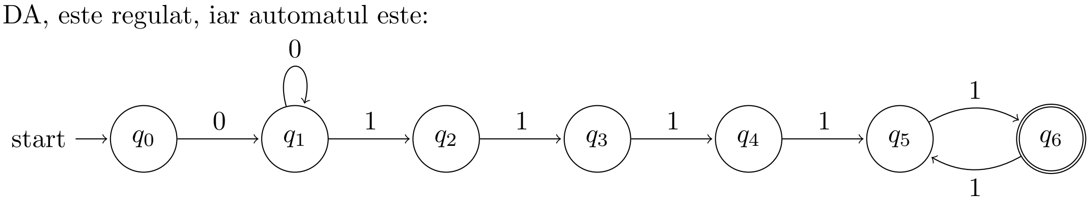
2. Exerciţii

(1) Pentru fiecare dintre limbajele următoare stabiliţi dacă este regulat, iar în caz afirmativ construiţi automate finite:

(a) $L = \{0^{k-1}1^{2l+1} \mid k, l \geq 2\}$

Soluţie:

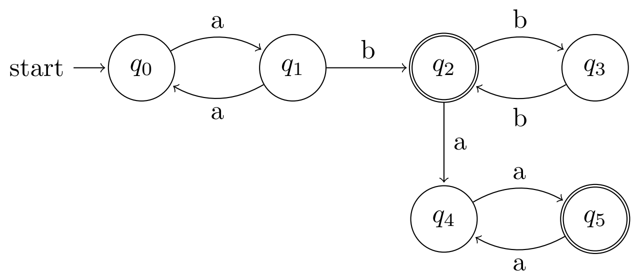
DA, este regulat, iar automatul este:



(b) $L = \{a^i b^j a^k \mid i$ impar, k e par şi j impar$\}$

Soluţie:

DA, este regulat, iar automatul este:

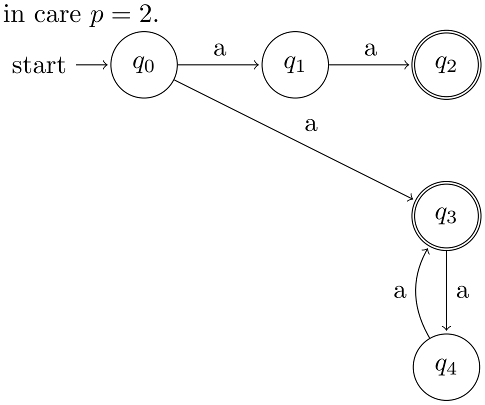


(c) $L = \{a^i b^j a^k \mid$ dacă i este impar, atunci k e par şi j impar; dacă i este par, atunci $i \leq 8, j \neq k\}$
Soluţie:

NU, nu este regulat.
Limbajul L poate fi rescris astfel:
$$L = \{a^i b^j a^k \mid i \text{ este impar, } k \text{ e par şi } j \text{ impar}\} \cup \{a^i b^j a^k \mid i \text{ este par,} i \leq 8, j \neq k\}$$
Primul limbaj din reuniune este regulat, fiind cel de la subpunctul aterior.
Acum ne vom ocupa de al doilea limbaj. Faptul ca i este un numar par, mai mic decat 8, nu ne incurca. Observam partea a doua, in care stim ca $j \neq k$. Daca acest limbaj ar fi regulat, stiind ca limbajele regulate sunt inchise la complementare, l-am putea complementa si am gasi un automat si pentru un limbaj cu $k = j$, ceea ce stim ca nu putem.
Deci al doilea limbaj din reuniune nu este regulat. Cele doua limbaje sunt disjuncte, deci L nu este regulat.

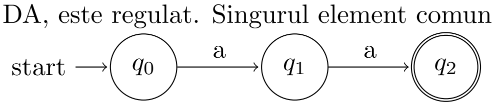
(d) $L = \{a^p \mid p \text{ prim}\} \cup \{a^p \mid p \text{ impar}\}$
Soluţie:

DA, este regulat. Singurul numar care este prim si nu este impar, adica se afla doar in prima multime, este 2. Asadar, vom reprezenta automatul pentru a doua multime, plus cazul particular in care $p = 2$.



(e) $L = \{a^p \mid p \text{ prim}\} \cap \{a^p \mid p \text{ par}\}$
Soluţie:

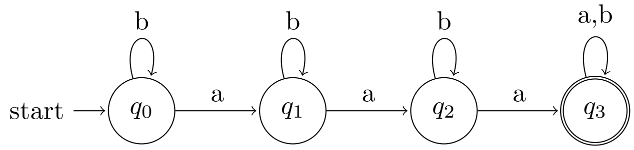
DA, este regulat. Singurul element comun al celor doua multimi este $\{a^2\}$.

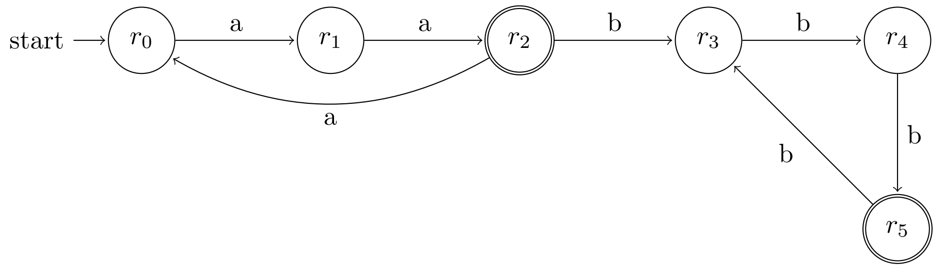


(2) Fie limbajele:

$$L_1 = \{w \in \{a,b\}^* \mid |w|_a \geq 3\}$$
$$L_2 = \{a^{3n+2} b^{3k} \mid n, k \geq 0\}$$

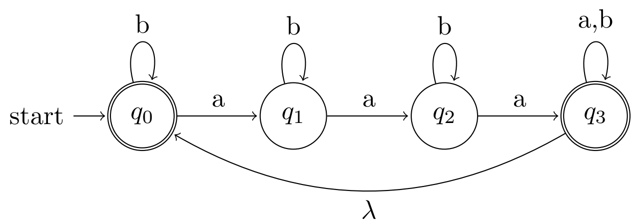
(a) Realizaţi câte un DFA pentru fiecare limbaj.
Soluţie:



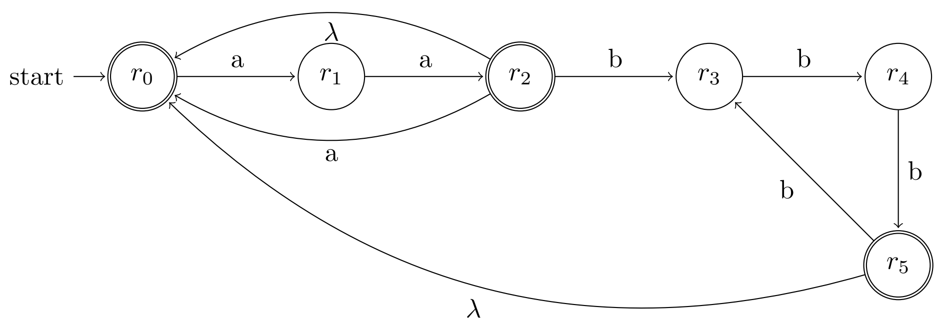


(b) Realizaţi automate pentru limbajele: $L_1^*, L_2^*, L_1^+, L_2^+, \overline{L_1}, \overline{L_2}$

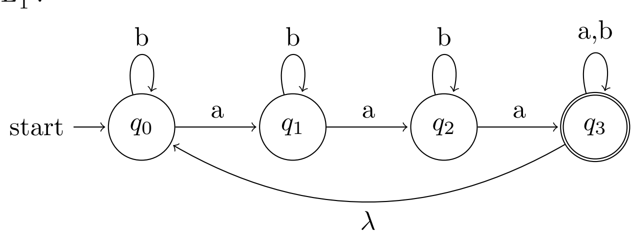
L_1^*:



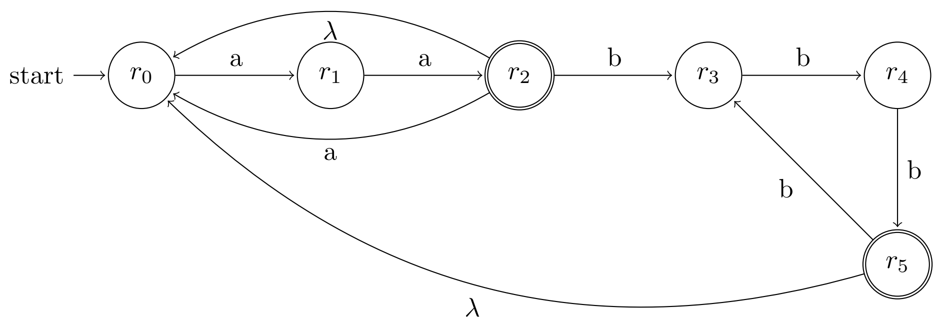
L_2^*:



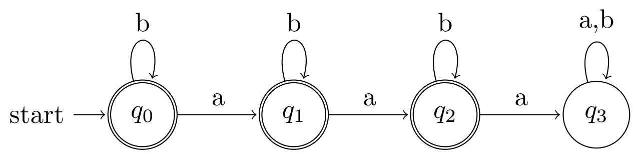
L_1^+:



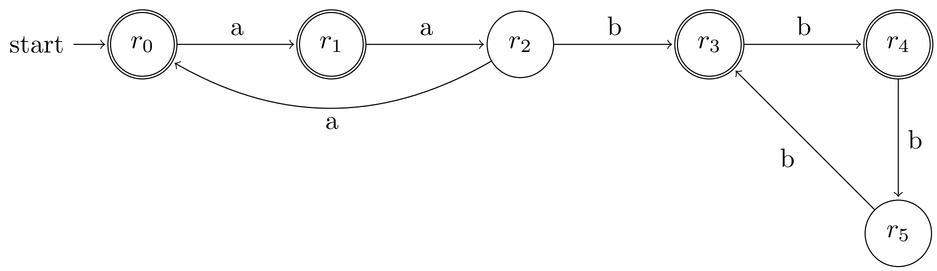
L_2^+:



$\overline{L_1}$:



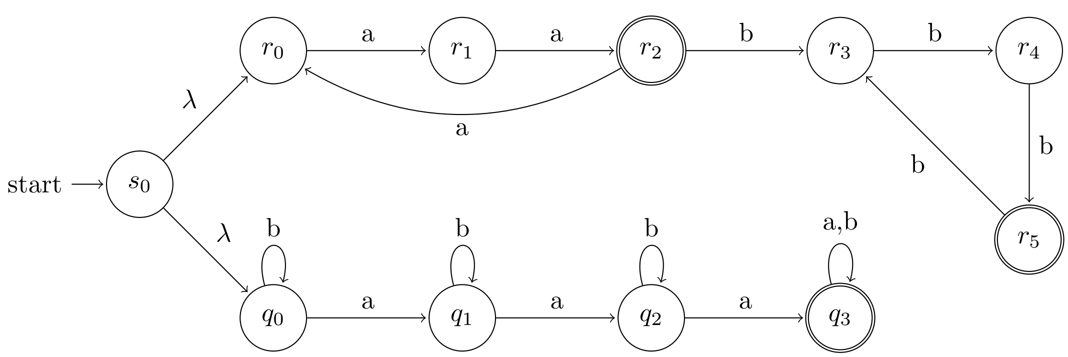
$\overline{L_2}$:



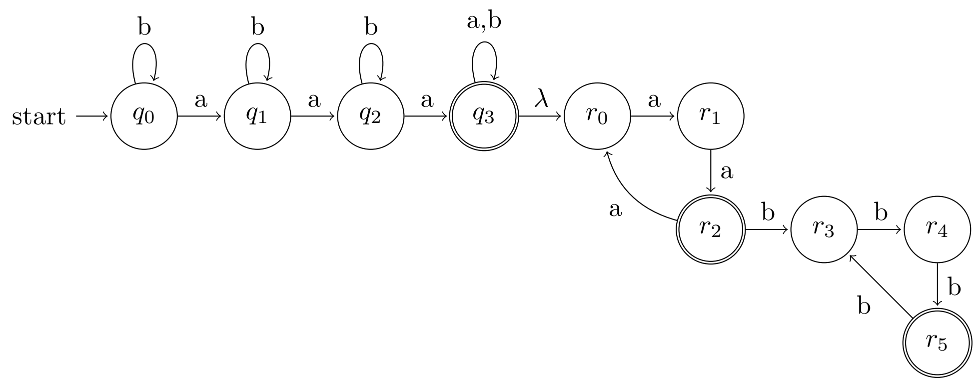
(c) Realizaţi automate pentru limbajele:
- $L_1 \cup L_2$ (λ-NFA)
- $L_1 \cdot L_2$ (λ-NFA)

Soluţie:

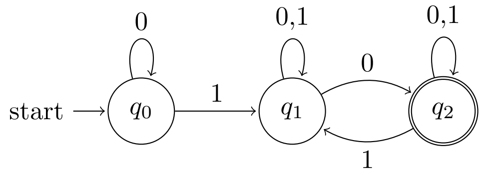
- $L_1 \cup L_2$ (λ-NFA)



- $L_1 \cdot L_2$ (λ-NFA)



(3) Transformaţi următorul într-un DFA echivalent:

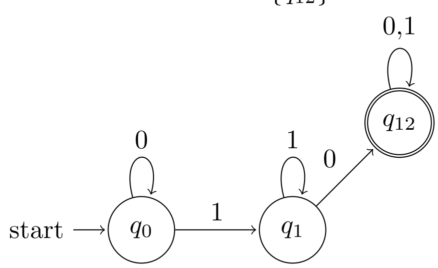


Soluţie:

Vom începe prin a calcula tabelul cu stările noului automat.

Stări	0	1
q_0	q_0	q_1
q_1	$q_{12} = \{q_1, q_2\}$	q_1
q_{12}	$q_{12} = \{q_1, q_2\}$	$q_{12} = \{q_1, q_2\}$

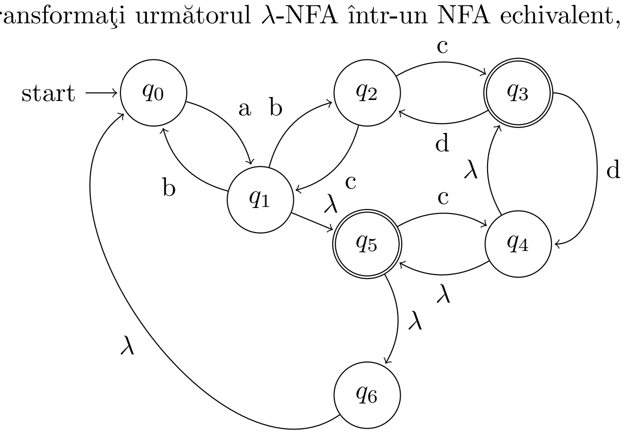
Noile stări finale sunt: $\{q_{12}\}$.



Putem observa că starea q_2 nu mai este accesibilă, deci ar putea fi eliminată

.

(4) Transformaţi următorul λ-NFA într-un NFA echivalent, iar apoi în DFA.



Soluţie:

Pasul 1: Determinăm λ-închiderile:

$$\langle q_0 \rangle = \{q_0\}$$
$$\langle q_1 \rangle = \{q_0, q_1, q_5, q_6\}$$
$$\langle q_2 \rangle = \{q_2\}$$
$$\langle q_3 \rangle = \{q_3\}$$
$$\langle q_4 \rangle = \{q_0, q_3, q_4, q_5, q_6\}$$
$$\langle q_5 \rangle = \{q_0, q_5, q_6\}$$
$$\langle q_6 \rangle = \{q_0, q_6\}$$

Stările finale în NFA vor fi $F' = \{q_1, q_3, q_4, q_5\}$.

Pasul 2: Realizăm tabelul de tranziţii:

stare	a	b	c	d	λ	λ^*
q_0	$\{q_1\}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_0\}$
q_1	$\emptyset$	$\{q_0, q_2\}$	$\emptyset$	$\emptyset$	$\{q_5\}$	$\{q_0, q_1, q_5, q_6\}$
q_2	$\emptyset$	$\emptyset$	$\{q_1, q_3\}$	$\emptyset$	$\emptyset$	$\{q_2\}$
q_3	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_2, q_4\}$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$	$\{q_0, q_3, q_4, q_5, q_6\}$
q_5	$\emptyset$	$\emptyset$	$\{q_4\}$	$\emptyset$	$\{q_6\}$	$\{q_0, q_5, q_6\}$
q_6	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_0\}$	$\{q_0, q_6\}$

Pasul 3: Determinăm tranziţiile de forma $\lambda^* \alpha \lambda^*$:

stare	$\lambda^* a \lambda^*$	$\lambda^* b \lambda^*$	$\lambda^* c \lambda^*$	$\lambda^* d \lambda^*$
q_0	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\emptyset$	$\emptyset$
q_1	$\{q_0, q_1, q_5, q_6\}$	$\{q_0, q_2\}$	$\{q_0, q_3, q_4, q_5, q_6\}$	$\emptyset$
q_2	$\emptyset$	$\emptyset$	$\{q_0, q_1, q_3, q_5, q_6\}$	$\emptyset$
q_3	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_0, q_2, q_3, q_4, q_5, q_6\}$
q_4	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\{q_0, q_3, q_4, q_5, q_6\}$	$\{q_0, q_2, q_3, q_4, q_5, q_6\}$
q_5	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\{q_0, q_3, q_4, q_5, q_6\}$	$\emptyset$
q_6	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\emptyset$	$\emptyset$

Pasul 4: Determinăm tranziţiile din NFA şi reprezntăm automatul:

stare	a	b	c	d
q_0 (starea iniţială)	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\emptyset$	$\emptyset$
$q_1 \in F'$	$\{q_0, q_1, q_5, q_6\}$	$\{q_0, q_2\}$	$\{q_0, q_3, q_4, q_5, q_6\}$	$\emptyset$
q_2	$\emptyset$	$\emptyset$	$\{q_0, q_1, q_3, q_5, q_6\}$	$\emptyset$
$q_3 \in F'$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_0, q_2, q_3, q_4, q_5, q_6\}$
$q_4 \in F'$	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\{q_0, q_3, q_4, q_5, q_6\}$	$\{q_0, q_2, q_3, q_4, q_5, q_6\}$
$q_5 \in F'$	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\{q_0, q_3, q_4, q_5, q_6\}$	$\emptyset$
q_6	$\{q_0, q_1, q_5, q_6\}$	$\emptyset$	$\emptyset$	$\emptyset$