

# LFA Laborator 3

Miriam Costan

Martie 2020

În următoarele două laboratoare vom face conversia unui  $\lambda - NFA$  la un  $DFA_{min}$ . Cu alte cuvinte vrem un program care primește ca intrare un  $\lambda - NFA$  și afișează  $DFA$ -ul echivalent cu un număr minim de stări.

## 0.1 Pași conversiei.

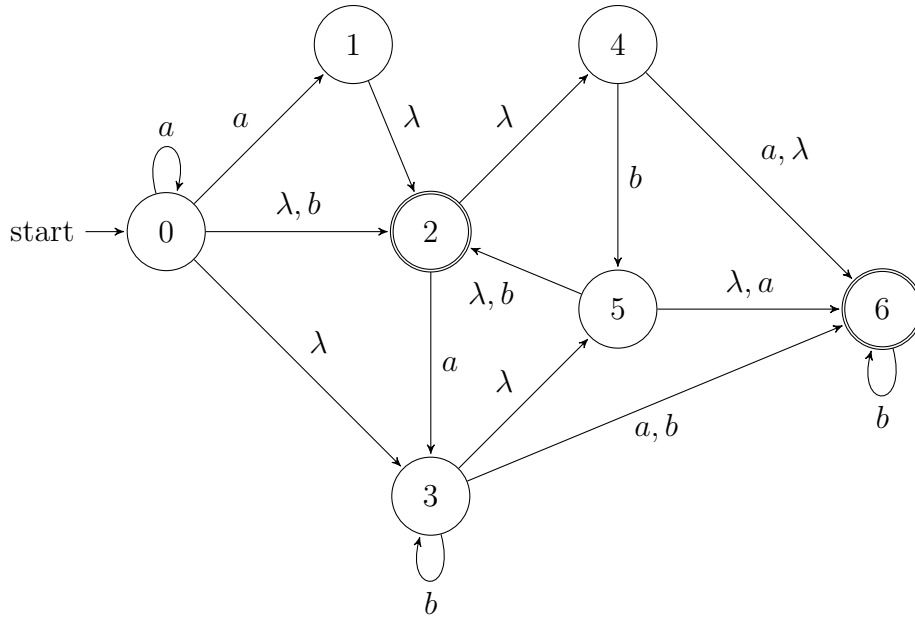
1. Laboratorul 3 - Conversie de la  $\lambda - NFA$  la  $NFA$ .
2. Laboratorul 3 - Conversie de la  $NFA$  la  $DFA$ .
3. Laboratorul 4 - Conversie de la  $DFA$  la  $DFA_{min}$ .

## 0.2 Punctajul pe cea de-a doua temă.

- 5 - Pentru oricare din cei 3 pași.
- 7 - Pentru oricare 2 din cei 3 pași.
- 10 - Pentru toți cei 3 pași.

## 1 $\lambda - NFA \rightarrow NFA$ .

Având dat un  $\lambda - NFA$ , vom construi  $NFA$ -ul echivalent prin eliminarea tranzițiilor de tip  $\lambda$ .



$\delta$	a	b	$\lambda$
0	{0, 1}	{2}	{2, 3}
1	{}	{}	{2}
2	{3}	{}	{4}
3	{6}	{3, 6}	{5}
4	{6}	{5}	{2, 6}
5	{6}	{2}	{2, 6}
6	{}	{6}	{}

### 1.1 Pasul 1. Calcularea $\lambda$ -inchiderii.

$\lambda$ -inchiderea( $\lambda^*$ ) unei stări  $q$  reprezintă mulțimea de stări în care se poate ajunge plecând din  $q$  cu 0 sau mai multe tranziții de tip  $\lambda$ .

**Observatie 1.** Orice stare face parte din propria sa  $\lambda$ -inchidere. Practic putem considera ca pentru orice stare există o  $\lambda$ -tranziție implicită către ea însăși.

**Observatie 2.**  $\lambda$ -inchiderea unei mulțimi de stări este egală cu reuniunea  $\lambda$ -inchiderii fiecărei stări din mulțime.

	$\lambda^*$
0	{0, 2, 3, 4, 5, 6}
1	{1, 2, 4, 6}
2	{2, 4, 6}
3	{2, 3, 4, 5, 6}
4	{4, 6}
5	{2, 4, 5, 6}
6	{6}

$\delta^*$	a	b
0	{0, 1, 2, 3, 4, 5, 6}	{2, 3, 4, 5, 6}
1	{2, 3, 4, 5, 6}	{2, 4, 5, 6}
2	{2, 3, 4, 5, 6}	{2, 4, 5, 6}
3	{2, 3, 4, 5, 6}	{2, 3, 4, 5, 6}
4	{6}	{2, 4, 5, 6}
5	{2, 3, 4, 5, 6}	{2, 4, 5, 6}
6	{}	{6}

## 1.2 Pasul 2. Calcularea functiei de tranzitie $\delta^*$ .

Cu ajutorul  $\lambda$ -inchiderii, putem calcula functia de tranzitie a *NFA*-ului pe care dorim sa il construim.

O stare  $q$  poate ajunge cu caracterul  $\alpha$  in starile ce rezulta dupa concatenarea la stanga si la dreapta cu  $\lambda^*$ . Cu alte cuvinte starile rezultate din calcularea  $\lambda^*\alpha\lambda^*$ .

Exemplu pentru caracterul

	$\lambda^*$	a	$\lambda^*$
0	{0, 2, 3, 4, 5, 6}	{0, 1, 3, 6}	{0, 1, 2, 3, 4, 5, 6}
1	{1, 2, 4, 6}	{3, 6}	{2, 3, 4, 5, 6}
2	{2, 4, 6}	{3, 6}	{2, 3, 4, 5, 6}
3	{2, 3, 4, 5, 6}	{3, 6}	{2, 3, 4, 5, 6}
4	{4, 6}	{6}	{6}
5	{2, 4, 5, 6}	{3, 6}	{2, 3, 4, 5, 6}
6	{6}	{}	{}

- Prima coloana reprezinta  $\lambda$ -inchiderea starii de pe linia respectiva.
- A doua coloana reprezinta reuniunea starilor accesibile cu caracterul  $a$  din fiecare din starile din  $\lambda$ -inchidere.
- A treia coloana reprezinta  $\lambda$ -inchiderea multimii de pe a doua coloana. Aceasta din urma va fi tranzitia cu caracterul  $a$  a *NFA*-ului construit.

## 1.3 Pasul 3. Calcularea starilor finale si initiale.

- Starea initiala ramane aceasi cu cea a automatului initial, in cazul nostru 0.
- Starile finale vor fi toate starile care contin o stare finala din automatul initial in  $\lambda$ -inchidere, in cazul nostru toate starile sunt in aceasta situatie.

## 1.4 Pasul 4. Eliminarea starilor redundante.

Doua stari sunt identice daca au tranzitiile sunt identice pentru orice caracter din alfabet si daca amandoua sunt sau nu sunt stari finale. In cazul nostru 1, 2, 5 sunt identice.

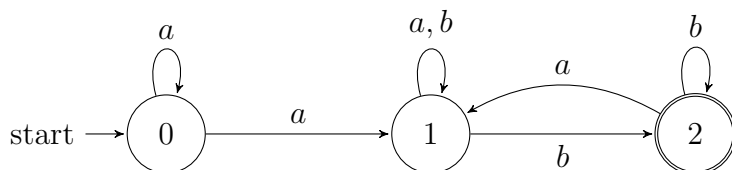
**Observatie 3.** Daca  $q$  si  $r$  sunt identice putem sa stergem starea  $r$  si sa inlocuim  $r$  cu  $q$  peste tot in tabelul de tranzitii.

$\delta^*$	a	b
0	{0, 1, 3, 4, 6}	{1, 3, 4, 6}
1	{1, 3, 4, 6}	{1, 4, 6}
3	{1, 3, 4, 6}	{1, 3, 4, 6}
4	{6}	{1, 4, 6}
6	{}	{6}

$\delta^*$	a	b
0	{0, 1, 3, 4, 6}	{1, 3, 4, 6}
1	{1, 3, 4, 6}	{1, 4, 6}
2	{1, 3, 4, 6}	{1, 4, 6}
3	{1, 3, 4, 6}	{1, 3, 4, 6}
4	{6}	{1, 4, 6}
5	{2, 3, 4, 5, 6}	{2, 4, 5, 6}
6	{}	{6}

## 2 $NFA \rightarrow DFA$ .

Avand dat un  $NFA$ , vom construi  $DFA$ -ul echivalent prin eliminarea nedeterminismului.



$\delta$	a	b
0	{0, 1}	{}
1	{1}	{1, 2}
2	{1}	{2}

### 2.1 Pasul 1. Eliminarea nedeterminismului.

Pornim cu o coada in care adaugam doar starea initiala  $q_0$ . Apoi pentru fiecare stare din coada  $q$  si fiecare caracter din alfabet  $\alpha$  facem urmatorul calcul:

daca  $\delta(q, \alpha) = q_{x_0}, \dots, q_{x_k}, k \geq 0$  atunci:

creem starea  $q_{x_0 \dots x_k}$  (poate fi si o stare care nu este compusa)

Daca noua stare formata  $q_{x_0 \dots x_k}$  nu a mai fost vizitata, atunci o adaugam in coada. Tranzitia acestei stari cu un caracter  $\alpha$  va fi reuniunea starilor accesibile cu caracterul  $\alpha$  din toate starile componente.

Repetam acest calcul pana cand coada devine vida.

$\delta^*$	a	b
0	01	
1	1	12
2	1	2
01	01	12
12	1	12

**Observatie 4.** Chiar daca numarul starilor poate creste exponential, asta nu se intampla de obicei in practica.

## 2.2 Pasul 2. Calcularea starilor initiale si finale.

- Starea initiala ramane aceasi cu cea a automatului initial, in cazul nostru 0.
- Starile finale vor fi toate starile care au in componenta o stare finala din automatul initial, in cazul nostru 2, 12.

## 2.3 Pasul 3. Redenumirea starilor.

Putem sa redenumim starile fara a afecta functionalitatea.

$\delta^*$	a	b
0	3	
1	1	4
2	1	2
3	3	4
4	1	4

