

LFA Laborator 2

Miriam Costan

Martie 2020

Definitie. Un automat finit este un 5-tuplu $\langle Q, \Sigma, \delta, q_0, F \rangle$, unde:

- Q reprezinta multimea finita a starilor.
- Σ reprezinta alfabetul.
- δ reprezinta functia de tranzitie. $\delta : Q \times \Sigma \rightarrow Q$, unde $\delta(x, \alpha) = y$ reprezinta faptul ca din starea x trecem in starea y la citirea caracterului α .
- q_0 starea initiala.
- $F \subset Q$, multimea starilor finale.

In laboratorul 2 vom scrie functia:

```
bool evaluate(automata, word) {  
    if(word is accepted by the automata)  
        return true;  
    else  
        return false;  
}
```

Cu alte cuvinte, vrem un program care primeste ca input un automat(DFA, NFA sau λ -NFA) si un set de cuvinte, si decide daca respectivele cuvinte sunt acceptate de automat sau nu.

Formatul datelor de intrare va fi dupa cum urmeaza:

- n = numarul de stari. Starile vor fi numerotate de la 0 la $n - 1$.
- m = numarul de caractere din alfabet.
- m caractere reprezentand alfabetul.
- q_0 = starea initiala.
- k = numarul de stari finale.
- k numere diferite intre 0 si $n - 1$ reprezentand starile finale.

- l = numărul de tranzitii.
- l tranzitii de tipul $x \alpha y$ reprezentand faptul ca din starea x mergem in starea y cu caracterul α

Vom oferi indicatii de implementare pentru fiecare din cele trei cazuri, plus exemple de testare.

1 DFA acceptor

1.1 Indicatii de implementare

1. Construim matricea de tranzitie δ astfel:

δ	α_0	α_1	α_2	α_3
q_0	q_x	q_z	N/A	N/A
q_1	q_v	q_y	N/A	q_x
q_2	q_w	N/A	N/A	N/A
q_3	N/A	q_u	N/A	q_z

2. Retinem la fiecare pas starea curenta. Cand citim un nou caracter din cuvânt, trecem la starea urmatoare corespunzator matricei de tranzitie.
3. Daca matricea de tranzitie nu are initializata celula care ne intereseaza δ (stare curenta, caracter curent), putem sa ne oprim, cuvântul nu poate fi acceptat de automat.
4. Ramane sa verificam atunci cand terminam de citit cuvântul, daca starea pe care ne-am oprit este finala.

1.2 Exemplu

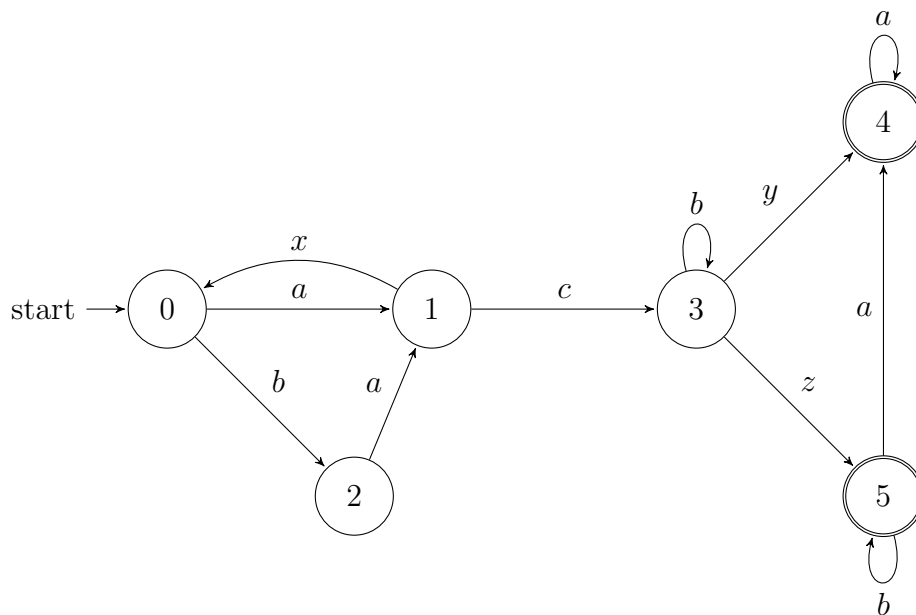
```

6
6
a b c x y z
0
2
4 5
11
0 a 1
0 b 2
1 c 3
1 x 0
2 a 1
3 b 3
3 y 4

```

3 z 5
 4 a 4
 5 a 4
 5 b 5

axbacbbzbbaaa - TRUE
 axccbya - FALSE
 axbac - FALSE
 bacy - TRUE
 bacyaaac - FALSE



2 NFA acceptor

2.1 Indicatii de implementare

1. Matricea de tranzitie δ se modifica astfel:

δ	α_0	α_1	α_2	α_3
q_0	$\{\}$	$\{\}$	$\{\}$	$\{\}$
q_1	$\{\}$	$\{\}$	$\{\}$	$\{\}$
q_2	$\{\}$	$\{\}$	$\{\}$	$\{\}$
q_3	$\{\}$	$\{\}$	$\{\}$	$\{\}$

In loc de o stare, matricea de tranzitii indica catre o multime de stari (posibil vida) in care putem sa ajungem la citirea fiecarui caracter.

2. In loc sa mai retinem starea curenta, retinem un set de stari curente in care putem ajunge.

3. La final este de ajuns ca una din starile in care ne-am oprit sa fie finala.

3 λ -NFA acceptor

3.1 Indicatii de implementare

1. Matricea de tranzitie δ se modifica astfel:

δ	α_0	α_1	α_2	λ
q_0	$\{\}$	$\{\}$	$\{\}$	$\{\}$
q_1	$\{\}$	$\{\}$	$\{\}$	$\{\}$
q_2	$\{\}$	$\{\}$	$\{\}$	$\{\}$
q_3	$\{\}$	$\{\}$	$\{\}$	$\{\}$

Introducem in matricea de tranzitii caracterul λ

2. De tinut cont ca λ nu citeste niciun caracter din cuvant pentru a il folosi.
3. la fiecare pas cand actualizam starile curente, trebuie sa tinem cont si de starile in care ajungem cu λ tranzitie.

3.2 Exemplu

- In fisierul din input λ va fi reprezentat cu ajutorul caracterului \$.
- NU exista λ - cicluri.

```

9
6
a b c x y z
0
2
7 8
16
0 a 1
0 a 2
0 b 2
1 $ 3
2 c 4
3 b 5
3 $ 4
4 a 5
4 b 6
5 $ 7
6 x 7

```

7 y 7
 7 x 8
 7 z 8
 8 y 6
 8 y 3

abxyyyxyby - TRUE
 bcax - TRUE
 bcbxxy - FALSE
 abyyxz - FALSE
 abyyxyx - TRUE

