

Multimi regulate

O **multime regulata** asupra unui alfabet Σ este o multime construita recursiv in urmatorul mod:

- \emptyset este o multime regulată asupra alfabetului Σ ;
- $\{e\}$ este o multime regulată asupra alfabetului Σ ;
- Daca $a \in \Sigma$ atunci $\{a\}$ este o multime regulata asupra alfabetului Σ ;
- Daca P și Q sunt multimi regulate atunci:
 - $P \cup Q$;
 - PQ ;
 - P^*, Q^*sunt multimi regulate asupra alfabetului Σ .
- Nimic altceva nu este o multime regulata.

Expresii regulate

O **expresie regulata** este definita recursiv:

- Φ este o expresie regulata care descrie multimea regulata Φ ,
- e este o expresie regulata care descrie multimea regulata $\{e\}$;
- $a \in \Sigma$ este o expresie regulata care descrie multimea regulata $\{a\}$;
- daca p, q sunt expresii regulate care descriu multimele regulate P , respectiv Q atunci :
 - $(p + q)$ sau $(p \mid q)$ este o expresie regulata care descrie multimea regulata $P \cup Q$;
 - (pq) este o expresie regulata care descrie multimea regulata PQ ;
 - $(p)^*$, $(q)^*$ sunt expresii regulate care descriu multimele regulate P^* , respectiv Q^* .
- nimic altceva nu este o expresie regulata.

Exemple

- ER 01 descrie $MR = \{01\}$
- ER 0^* descrie $MR = \{0\}^*$
- ER $(0+1)^*011$ descrie

$MR = \{w \in \{0,1\}^* \mid w \text{ se termina cu } 011\}$

Operatii asupra ER

*

.

+ (|)



Scade prioritatea

Proprietati ale expresiilor regulate

1. $a + b = b + a$

2. $a + (b+g) = (a+ b) + g$

3. $a(bg) = (ab)g$

4. $a(b +g) = ab + ag$

5. $(a+ b)g = ag + bg$

6. $a + a = a$

7. $ae = ea = a$

8. $\Phi a = a\Phi = \Phi$

9. $e^* = e$

10. $\Phi^* = e$

11. $a^* = a + a^*$

12. $a + e = e + a = a?$

13. $a + \Phi = \Phi + a = a$

14. $(a^*)^* = a^*$

15. $(a^*b^*)^* = (a+b)^*$

Ecuatii in expresii regulate

Ecuatia $x = ax + b$, cu a , b , x expresii regulate are solutia minimala $x = a^*b$

Multimi regulate si gramatici regulate

Un limbaj L este o multime regulata $\Leftrightarrow L$ este un limbaj regulat.

Φ , $\{e\}$, $\{a\}$ pentru orice $a \in \Sigma$ sunt gramatici regulate:

- $G = (\{S\}, \Sigma, \Phi, S)$ este o gramatica regulata care genereaza $L(G) = \Phi$
- $G = (\{S\}, \Sigma, \{S \rightarrow e\}, S)$ este o gramatica regulata care genereaza $L(G) = \{e\}$
- $G = (\{S\}, \Sigma, \{S \rightarrow a\}, S)$ este o gramatica regulata care genereaza $L(G) = \{a\}$ pentru orice $a \in \Sigma$

Exemplu

$$G = (\{A, B, C\}, \{0, 1\}, P, A)$$

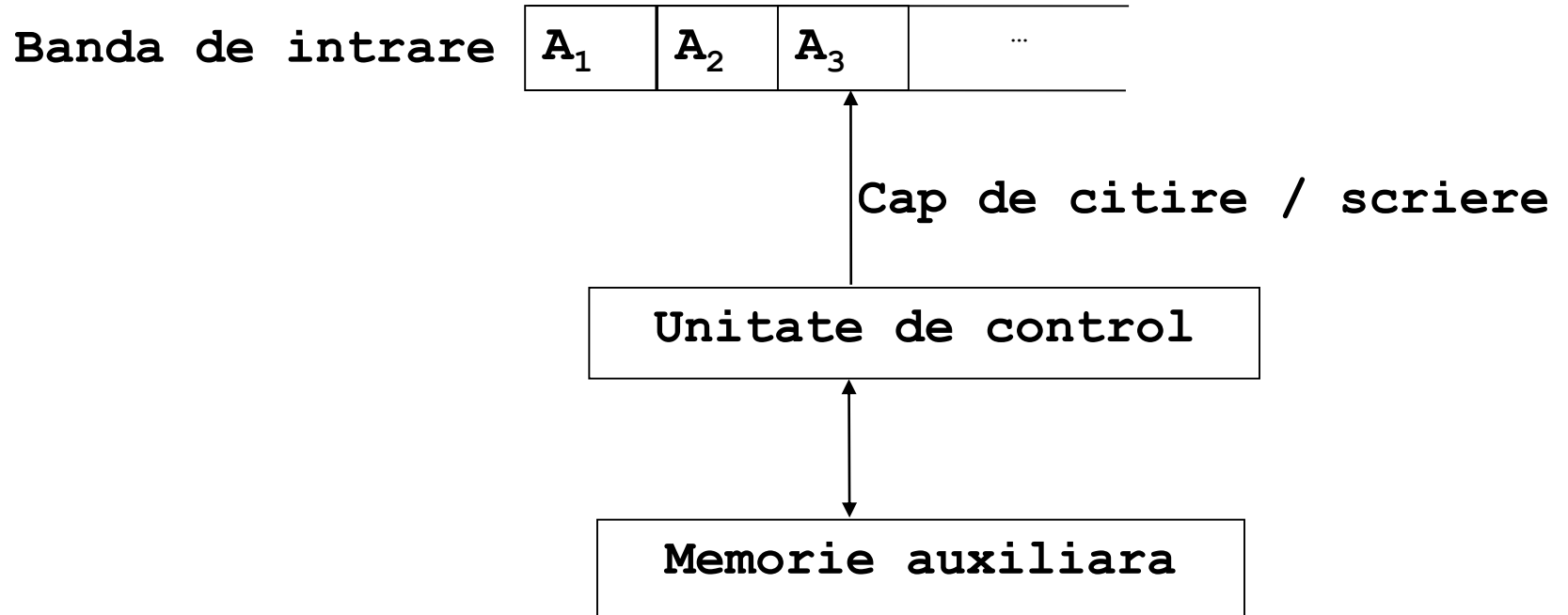
P:

$$A \rightarrow 0A \mid 1A \mid 0B$$

$$B \rightarrow 1C$$

$$C \rightarrow 1$$

Acceptoare



Acceptoare

O **miscare** a acceptorului:

- Modificarea simbolului curent de pe banda de intrare

Si/sau

- Deplasarea cu o pozitie a capului de citire/scriere (stanga/dreapta) sau mentinera capului pe aceeaasi pozitie

Si/sau

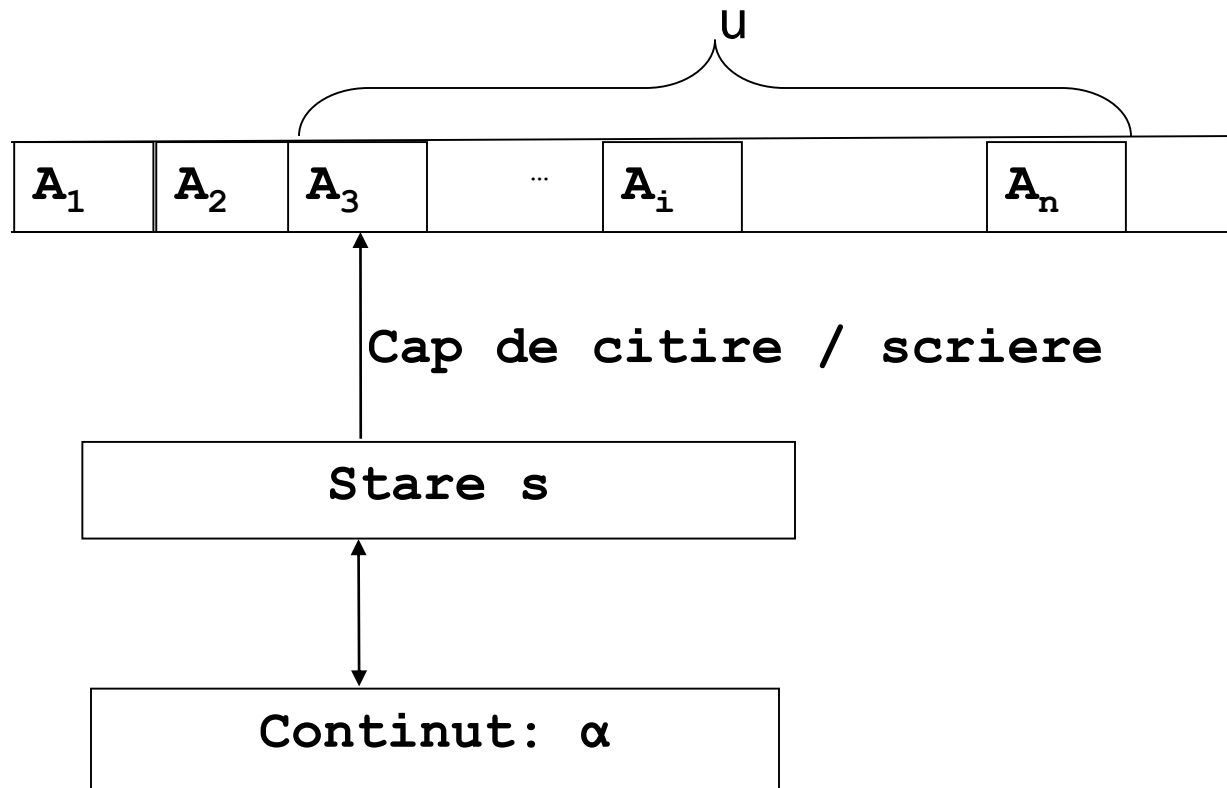
- Memorarea unei informatii in memoria auxiliara

Si/sau

- Modificarea starii unitatii de control

Configuratie

Configuratie: $q_i = (s, u, \alpha)$



Miscarea acceptorului

O **miscare** a acceptorului: $q_i \vdash\!\!-\! q_j$

Sucesiune de miscari ale acceptorului:
 $\vdash\!\!-\!^*$, $\vdash\!\!-\!+$

Configuratie initiala

Configuratie finala

Acceptoare

Un **sir w** este **acceptat** de un acceptor daca
configuratie initiala \vdash^* configuratie finala

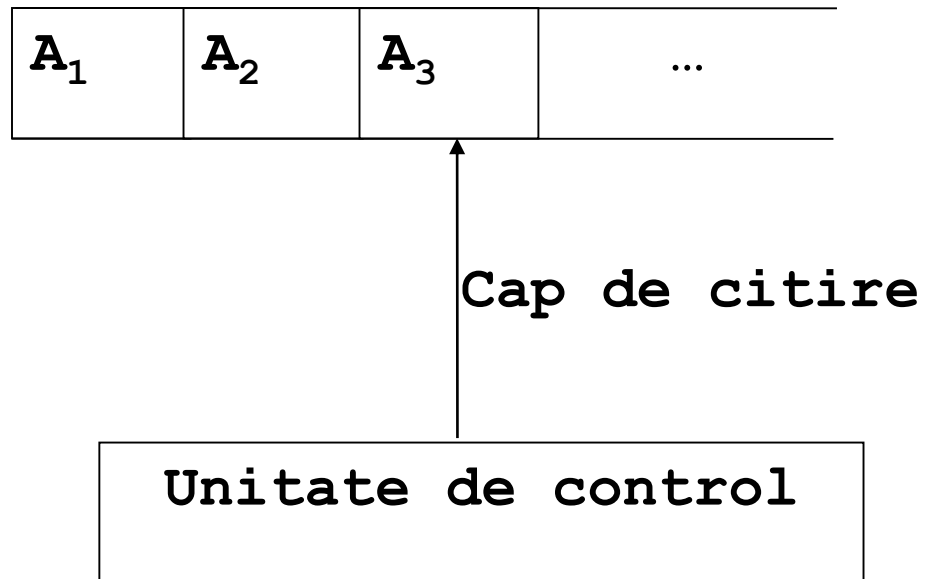
Tipuri de acceptoare:

- acceptoare deterministe
- acceptoare nedeterministe

Gramatici - acceptoare

Gramatici	Acceptoare
Gramatici fara restrictii	Masina Turing
Gramatici dependente de context	Automate liniar marginite
Gramatici independente de context	Automate cu stiva
Gramatici regulate	Automate finite

Automate finite



Automate finite nedeterministe

$$AF = (Q, \Sigma, m, q_0, F)$$

- Q este multimea finita a starilor;
- Σ este o multime finita numita alfabet de intrare;
- $m : Q \times (\Sigma \cup \{e\}) \rightarrow P(Q)$ este functia partiala a starii urmatoare (se mai numeste si functie de tranzitie);
- $q_0 \in Q$ stare de start;
- $F \subseteq Q$ este o multime de stari numita multimea starilor finale (sau multimea starilor de acceptare).

Automate finite nedeterministe

- **Tranzitie:** $m(q,a)$

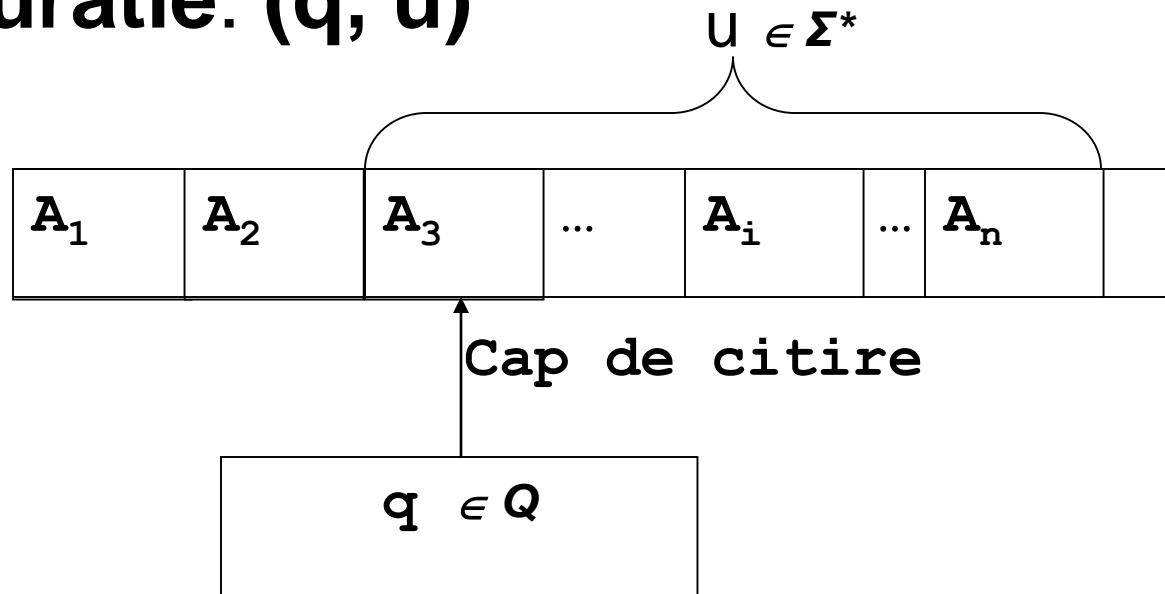
$$r \in m(q, a) \quad r, q \in Q, a \in \Sigma$$

- **e-tranzitie:** $m(q,e)$

$$r \in m(q, e) \quad q, r \in Q$$

Configuratie

- Configuratie: (q, u)



- configuratie initiala (q_0, w) , $q_0 \in Q$, starea initiala, $w \in \Sigma^*$
- configuratie finala (q_f, e) , $q_f \in F$.

Miscarea automatului

Miscarea automatului este specificata prin relatia : \vdash

- $(q_i, au) \vdash (q_j, u)$
 $q_i, q_j \in Q$ si $a \in \Sigma, u \in \Sigma^*, q_j \in m(q_i, a)$
- $(q_i, au) \vdash (q_j, au)$
 $q_i, q_j \in Q$ si $a=e, u \in \Sigma^*, q_j \in m(q_i, e)$

Automate finite - acceptare

$$\textbf{AF } M = (Q, \Sigma, m, q_0, F)$$

M accepta $w \in \Sigma^*$ daca:

$$(q_0, w) \vdash^{*} (q_f, e)$$

Limbajul acceptat de M:

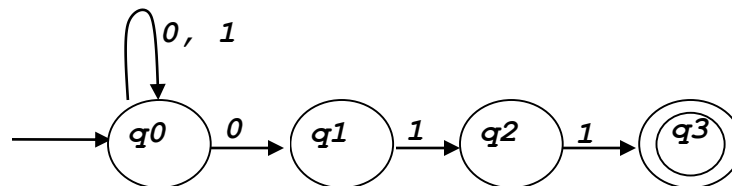
$$L(M) = \{w \in \Sigma^* \mid (q_0, w) \vdash^{*} (q_f, e)\}$$

Reprezentarea automatelor finite

Tabela de tranzitie

	<i>simbol de intrare</i>	
<i>stare</i>	<i>0</i>	<i>1</i>
<i>q0</i>	$\{q0, q1\}$	$\{q0\}$
<i>q1</i>	–	$\{q2\}$
<i>q2</i>	–	$\{q3\}$
<i>q3</i>	–	–

Graf de tranzitie



Automate finite deterministe

$$AF = (Q, \Sigma, m, q_0, F)$$

$$m : Q \times \Sigma \rightarrow Q$$

pentru care sunt satisfăcute următoarele restricții:

- 1. nu exista e-tranzitii;
- 2. pentru $\forall (q, a) \in Q \times \Sigma$ este definita cel mult o tranzitie.

Automate finite

Urmatoarele afirmatii sunt echivalente:

- L este multime regulata
- L este descris de o expresie regulata
- L este limbaj regulat
- L este limbaj acceptat de AFN
- L este limbaj acceptat de AFD

Exemplu

$$G = (\{A, B, C\}, \{0, 1\}, P, A)$$

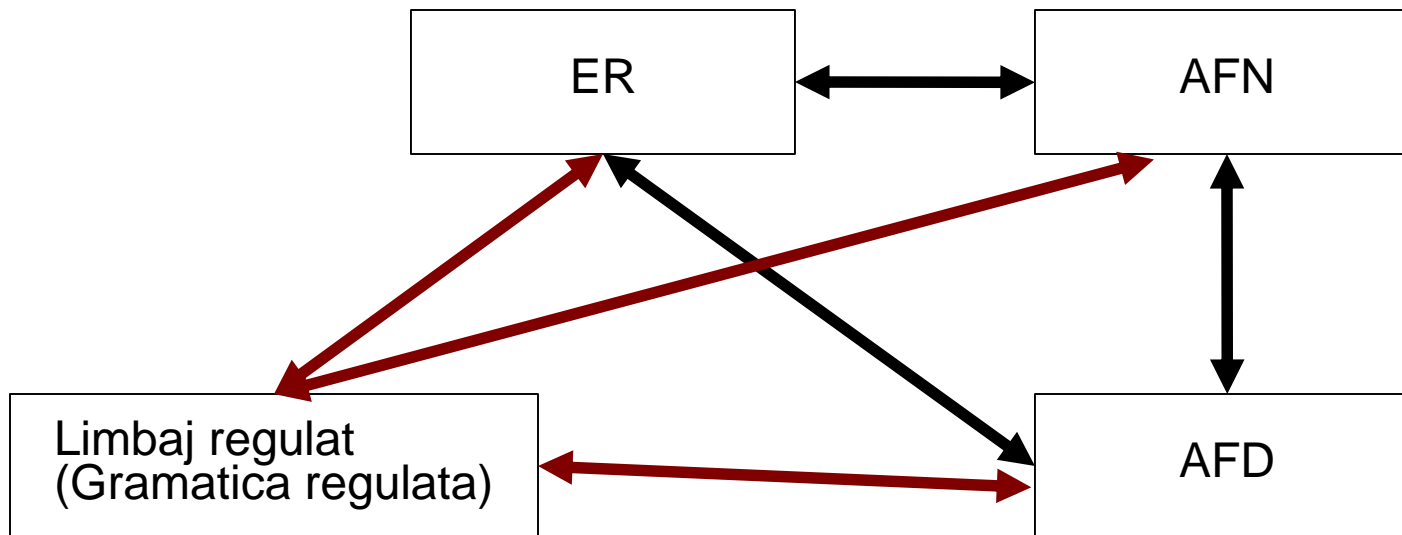
P:

$$A \rightarrow 0A \mid 1A \mid 0B$$

$$B \rightarrow 1C$$

$$C \rightarrow 1$$

Echivalenta ER, AFN, AFD, LR

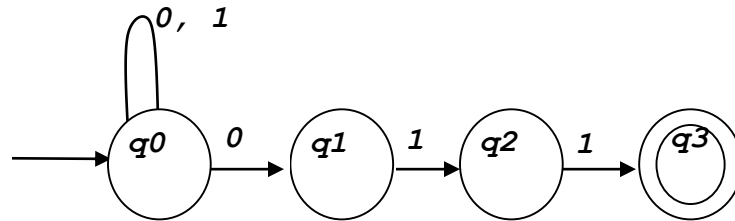


LR \rightarrow AFN

$S \rightarrow 0S \mid 0A \mid 1S$

$A \rightarrow 1B$

$B \rightarrow 1$



LR \rightarrow AFD

$S \rightarrow 0S \mid 0A \mid 1S$

$A \rightarrow 1B$

$B \rightarrow 1$