

Automate

- Recapitulare, Exemple, Aplicatii
- Translatoare
- Masini Turing

Automat finit: model fizic

banda de intrare



cap
citire-scriere



directie de deplasare



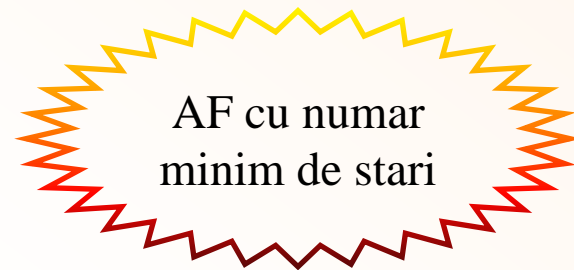
stari

Automat finit: model matematic

- Un *automat finit* este un ansamblu

$$M = (Q, \Sigma, \delta, q_0, F) :$$

- Q – alfabetul starilor
- Σ – alfabet de intrare
- $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ functie de tranzitie
- $q_0 \in Q$ - stare initială
- $F \subseteq Q$ multimea stărilor finale



Automat push-down (APD)

banda de intrare



cap de
citire

directie de deplasare



stari

varful stivei

z0

stiva

Automat push-down (APD)

$$M = (Q, \Sigma, \Gamma, \delta, q_o, Z_o, F)$$

- Q alfabetul stărilor;
- Σ alfabetul de intrare;
- Γ alfabetul memoriei stivă;
- $q_o \in Q$ stare inițială;
- $Z_o \in \Gamma$ simbolul de start al memoriei stivă;
- $F \subseteq Q$ mulțimea stărilor finale;
- $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow \mathcal{P}_o(Q \times \Gamma^*)$ funcția de tranziție
 - are ca valori submulțimi finite din $Q \times \Gamma^*$ (posibil mulțimea vidă)

Transformarea acceptarii dupa criteriul stivei vide in acceptare dupa criteriul starii finale

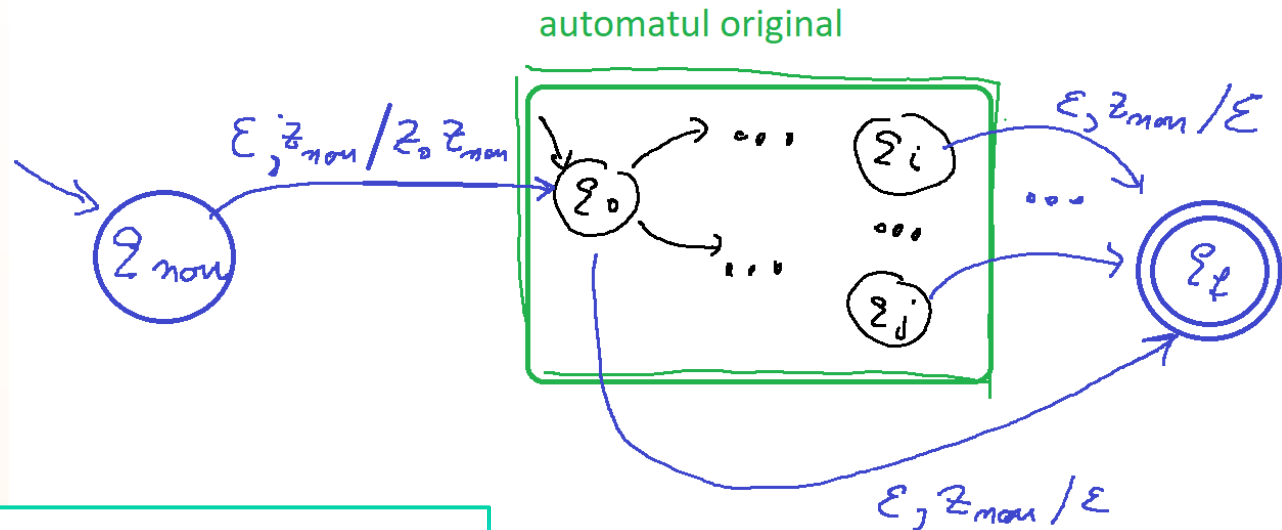
Fie: M – APD care accepta L dupa criteriul stivei vide

Dorim sa construim M' care accepta L dupa criteriul starii finale

M' – contine tot ce contine M , dar mai adaugam:

- q_{nou} o noua stare initiala
- q_f o noua starea finala
- Z_{nou} noul simbol initial al stivei

Adaugam si tranzitiile corespunzatoare noilor stari, asa cum este schitat mai jos:



In mod analog se poate face transformarea acceptarii dupa criteriul starii finale in acceptare dupa criteriul stivei vide.

Translator finit

$$M = (Q, \Sigma, D, \delta, q_o, F)$$

- Q alfabetul stărilor;
- Σ alfabetul de intrare;
- D alfabetul de iesire;
- $q_o \in Q$ stare inițială;
- $F \subseteq Q$ mulțimea stărilor finale;
- $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}_0(Q \times D^*)$
mulțimea partilor finite

banda de intrare

a1	a2	a3	...	a _n	
----	----	----	-----	----------------	--

cap de
citire

directie de deplasare

**Translator
finit**

UC

stari

cap de
scriere

banda de iesire

Translator finit

Exemplu:

$$M = (Q, \quad \Sigma, \quad D, \quad \delta, \quad q_o, \quad F)$$

$$M = (\{q_0, q_1\}, \quad \{a\}, \{b\} , \quad \delta , \quad q_0 , \quad \{q_1\})$$

$$\delta(q_0, a) = \{q_1, \{b\}\}$$

$$d(q_1, \varepsilon) = \{q_1, \{b\}\}$$

Translatarea definita de M:

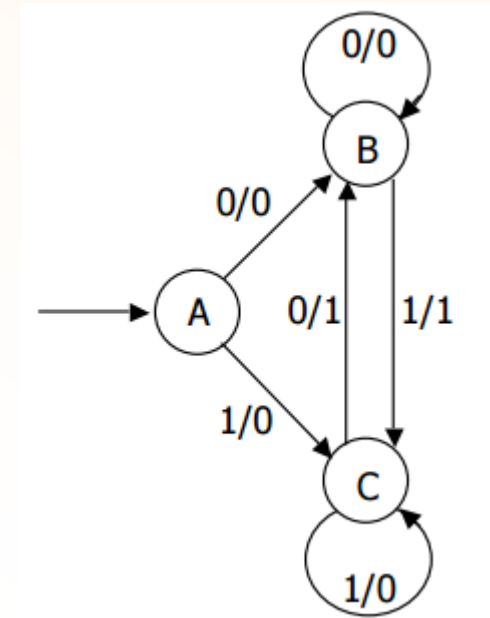
$$T(M) = \{ (x, y) \mid x \in \Sigma^* , y \in D^* , (q_0, x, \varepsilon) \vdash^* (q, \varepsilon, y), q \in F \}$$

Alte automate

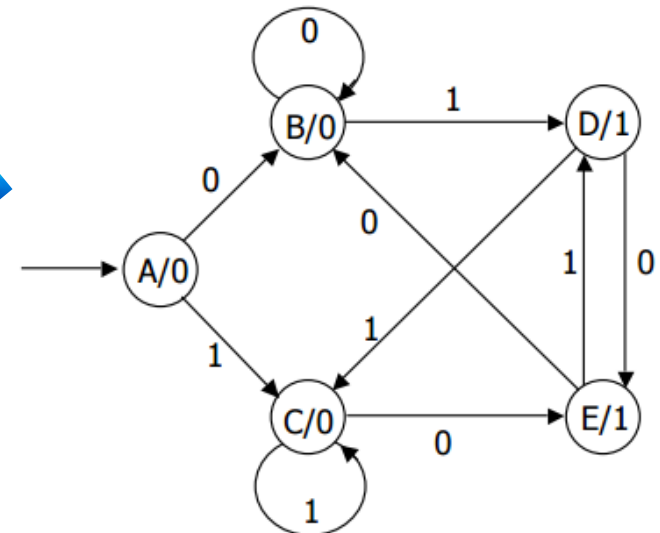
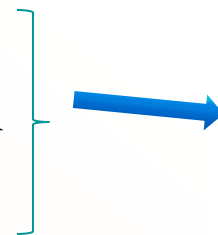
In literatura mai gasim:

(intre multe alte tipuri)

- **Automatele Mealy:** la fiecare tranzitie se produce un simbol de ieşire.



- **Automatele Moore:** fiecărei stări (intrare într-o stare) i se asociază un simbol de ieşire.



Translator push-down

$$M = (Q, \Sigma, \Gamma, D, \delta, q_0, Z_0, F)$$

- Q alfabetul stărilor;
- Σ alfabetul de intrare;
- Γ alfabetul memoriei stivă;
- D alfabetul de iesire;
- $q_0 \in Q$ stare inițială;
- $Z_0 \in \Gamma$ simbolul de start al memoriei stivă;
- $F \subseteq Q$ mulțimea stărilor finale;
- $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow \mathcal{P}_0(Q \times \Gamma^* \times D^*)$
multimea partilor finite

Translator push-down

banda de intrare



cap de citire

directie de deplasare



stari

varful stivei

Z0

stiva

cap de scriere

banda de iesire

Translator push-down

$$M = (Q, \Sigma, \Gamma, D, \delta, q_o, Z_o, F)$$

$$Q = \{q\}$$

$$\Sigma = \{a, +, *\}$$

$$\Gamma = \{E, +, *\}$$

$$D = \{a, +, *\}$$

$$q_o = q$$

$$Z_o = E$$

$$\delta(q, a, E) = \{(q, \varepsilon, a)\}$$

$$\delta(q, +, E) = \{(q, EE+, \varepsilon)\}$$

$$\delta(q, *, E) = \{(q, EE*, \varepsilon)\}$$

$$\delta(q, \varepsilon, +) = \{(q, \varepsilon, +)\}$$

$$\delta(q, \varepsilon, *) = \{(q, \varepsilon, *)\}$$

Considerand criteriul stivei vide,
descrieti translatarea pe care acesta o defineste .

... am lucrat
si cu alte
translatoare

Vezi:

LL(1)

LR(*)

Ne reamintim: Analizorul LL(1)

- Automat: (α, β, Π)
 - banda de intrare: α
 - stiva β (stiva de lucru)
 - banda de iesire $\Pi \Rightarrow$ sirul regulilor de productie
- config. initiala: $(w$, S, $\varepsilon)$$$
- config. finala: $($, $$, \Pi)$$
- tranzitii
 - push $(\mathbf{a}x$, $\mathbf{A}\beta$, $\Pi) \vdash (\mathbf{a}x$, $\alpha\beta$, $\Pi\mathbf{i})$ dc.: $M(A, a) = (\alpha, i)$$$
 - pop $(\mathbf{a}x$, $\mathbf{a}\beta$, $\Pi) \vdash (x$, β , $\Pi)$$$
 - acc $($, $$, $\Pi) \vdash \text{acc}$$$
 - err in celelalte cazuri

Automatul LL(1) ca translator push-down (modificat)

Translatorul push-down modificat este:

[Moldovan]

$$M = (\{q\}, \Sigma, \Gamma, D, \delta, q, S, \emptyset)$$

Σ - alfabet de intrare și este același cu alfabetul Σ din G ;

D - alfabet de ieșire, $D = \{1, 2, \dots, m\}$

$i \in D$ reprezintă nr. de ordine al producțiilor din P , $i: A \rightarrow \alpha$, $i = \overline{1, m}$

Γ - alfabetul memoriei push-down $\Gamma = N \cup \Sigma$

q - starea internă (stare inițială)

S - simbolul de start în memoria push-down, $S \in \Gamma$

δ - funcția de tranziție modificată

$$\delta: \Sigma \times \Gamma \rightarrow P(\Gamma^* \times (D \cup \{\varepsilon\}) \times \{0, 1\})$$

$$\delta(a, a) = \{(\varepsilon, \varepsilon, 1)\} \quad \forall a \in \Sigma$$

$$\delta(a, A) = \{(x, i, 0)\}, \quad a \in \Sigma, A \in N$$

dacă $(\exists)i: A \rightarrow x$ și dacă $a \in \varphi(x)$, sau

dacă $\varepsilon \in \varphi(x) \Rightarrow a \in \varphi(A)$

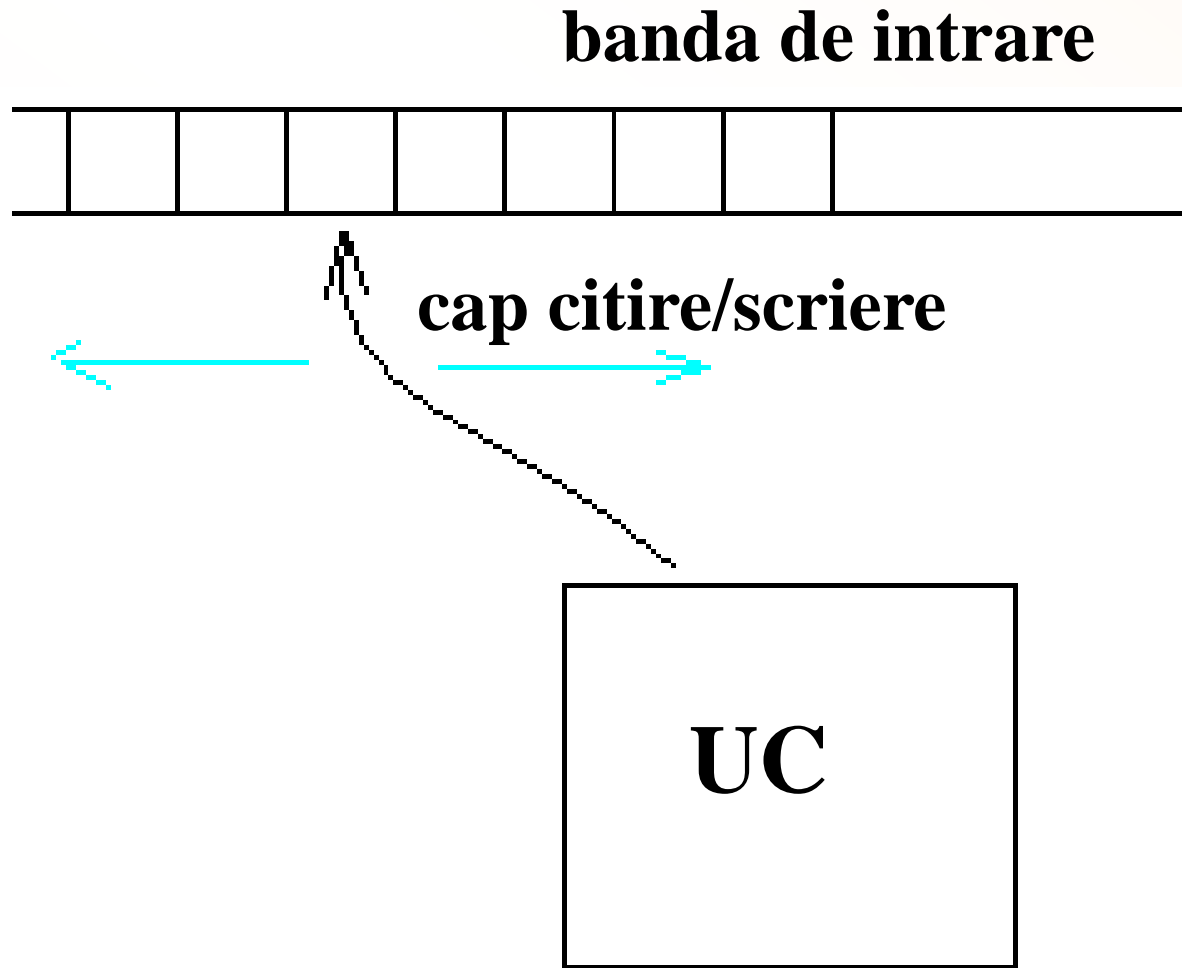
În celelalte cazuri $\delta(.,.) = \emptyset$

1 : semnifică înaintarea benzii cu o poziție;

0 : semnifică staționarea benzii.

Limbajul acceptat
se definește după
criteriul stivei vide.

Masini Turing



- infinita
- finita la stanga
- ...

Masini Turing

O miscare a masinii Turing consta din:

- se schimba starea
- se inlocuieste simbolul curent de pe banda de intrare
- capul citire/scriere se muta cu o pozitie la stanga sau la dreapta

Masina Turing cu banda infinita

O masina Turing este: $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$

- Q – multime finita de stari
- Γ – multimea simbolurilor benzii
- $\#$ - un simbol din Γ , numit simbolul *blanc*
- Σ – o submultime a lui $\Gamma - \{\#\}$
- δ – este functia de tranzitie

$$\delta: Q \times \Gamma \rightarrow \mathcal{P}(Q \times \Gamma \times \{L, R\})$$

- q_0 – starea initiala
- $F \subset Q$ – multimea starilor finale

Masina Turing cu banda infinita

- configuratie

$\alpha_1 q \alpha_2$ $\alpha_1 \alpha_2 \in \Gamma^*$, separate de capul de citire
pana la cel mai din stanga/dreapta simbol ne-blank

- tranzitie

$(p, Y, L) \in \delta(q, X_i)$

$X_1 X_2 \dots X_{i-1} q X_i X_{i+1} \dots X_n \vdash X_1 X_2 \dots X_{i-2} p X_{i-1} Y X_{i+1} \dots X_n$

$(p, Y, R) \in \delta(q, X_i)$

$X_1 X_2 \dots X_{i-1} q X_i X_{i+1} \dots X_n \vdash X_1 X_2 \dots X_{i-1} Y p X_{i+1} \dots X_n$

- limbaj acceptat

$\{w \in \Sigma^* \mid q_0 w \vdash \alpha_1 q \alpha_2, q \in F, \alpha_1 \alpha_2 \in \Gamma^*\}$

Exemplu: masina Turing

Funcția de tranziție

0011 ?

	Σ		$\Gamma - \Sigma$			
	0	1	X	Y	#	
q₀	(q ₁ ,X,R)			(q ₃ ,Y,R)		0
q₁	(q ₁ ,0,R)	(q ₂ ,Y,L)		(q ₁ ,Y,R)		0
q₂	(q ₂ ,0,L)		(q ₀ ,X,R)	(q ₂ ,Y,L)		0
q₃				(q ₃ ,Y,R)	(q ₄ ,# ,R)	0
q₄						1

Masini Turing

- Masina Turing cu o singura banda
versus Masina Turing cu mai multe benzi
O mașină Turing cu k benzi
nu este mai puternică decât o mașină Turing standard
- Mașină Turing deterministă (MTD)
versus mașină Turing nedeterministă (MTND)
Cele două sunt computațional echivalente,
adică orice MTND se poate transforma într-o MTD
(și invers).

Masini Turing

Teza lui Church

- Logicianul Alonzo Church a emis ipoteza că mașina Turing este modelul cel mai general de calcul care poate fi propus.
 - mașina Turing este la fel de puternică ca orice alt model de calcul
 - nu înseamnă că poate calcula la fel de repede ca orice alt model de calcul, ci că poate calcula aceleași lucruri
- Acest enunț nu este demonstrabil în sens matematic.

Dacă avem un model de calcul, putem defini precis ce înțelegem prin complexitate:

- **Timpul** de calcul pentru un șir dat la intrare: este numărul de mutări făcut de mașina Turing înainte de a intra în starea ``terminat";
- **Spațiul** consumat pentru un șir de intrare: este numărul de căsuțe de pe bandă pe care algoritmul le folosește în timpul execuției sale.

Potrivirea sirurilor de caractere (pattern matching) folosind AF

Exercitiu:

Gasiti subsecventa:

ACACAGA

in textul:

ACACACAGA

