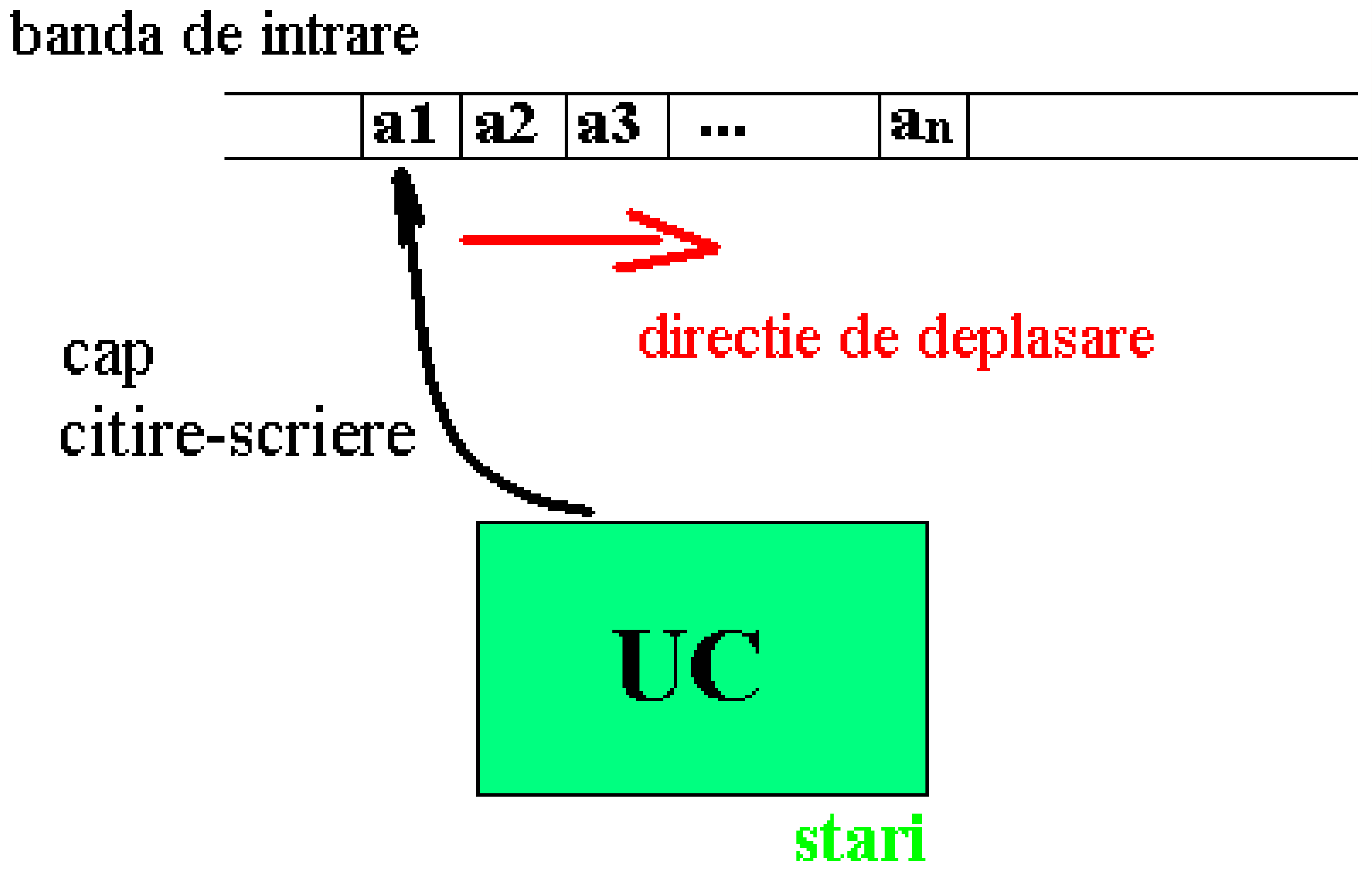
Automate

- Recapitulare, Exemple, Aplicatii
- Translatoare
- Masini Turing

Automat finit: model fizic



Automat finit: model matematic

- Un ***automat finit*** este un ansamblu

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

AF cu numar minim de stari

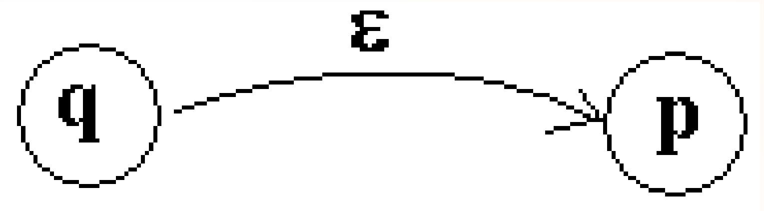
- Q – alfabetul starilor
- Σ – alfabet de intrare
- $\delta : Q x \Sigma \rightarrow \mathcal{P}(Q)$ functie de tranzitie
- $q_0 \in Q$ - stare initialã
- $F \subseteq Q$ multimea stãrilor finale

Automate finite cu ε-miscari

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

- $\delta : Qx(\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Q)$ functia de tranzitie

Putem avea si ε-tranzitii
(automate cu ε-tranzitii)

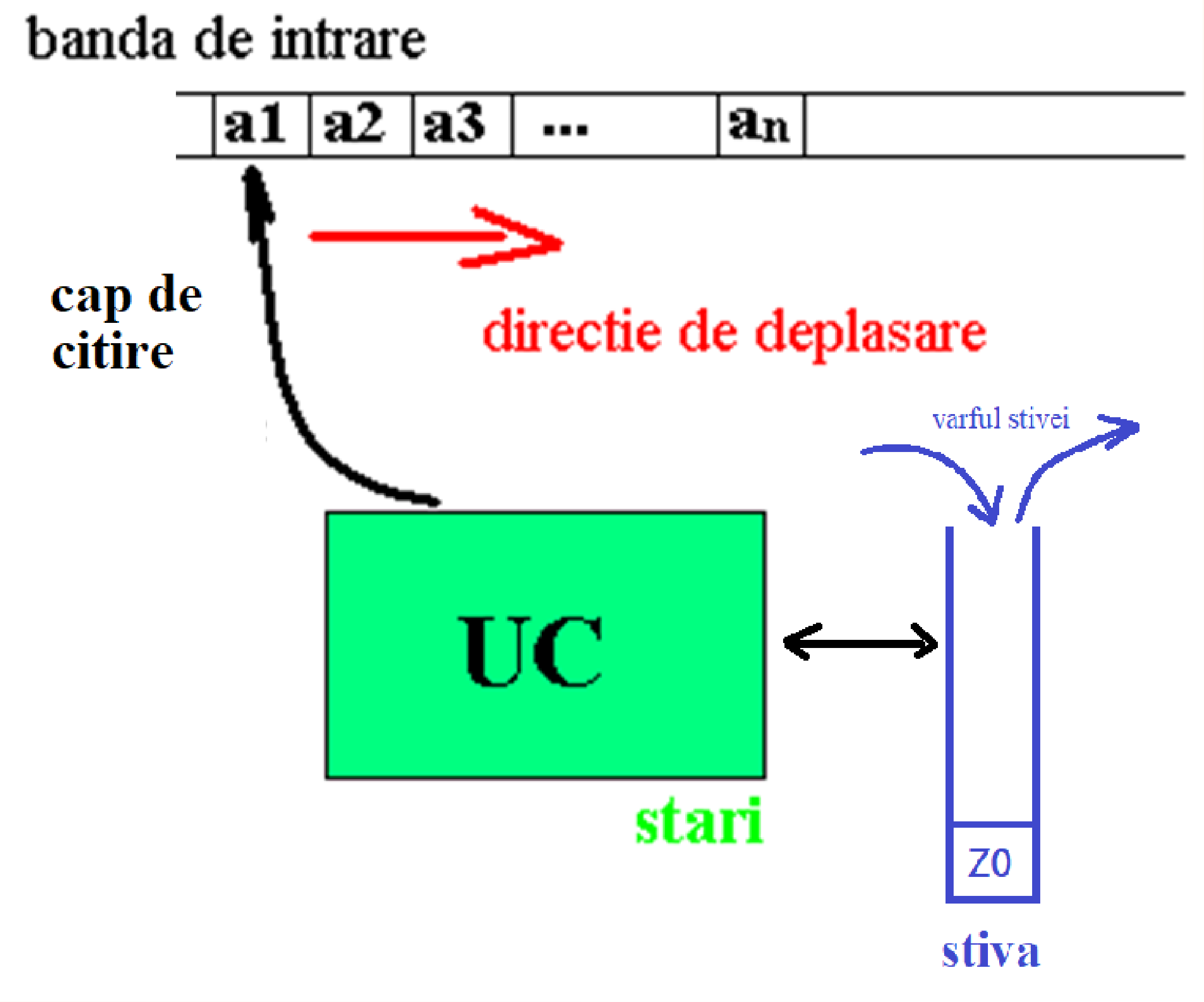


Teorema:

Pentru orice automat finit cu ε-miscari
exista un automat finit echivalent.

Obs. *Conform def., automatele finite sunt fara ε-miscari*

Automat push-down (APD)



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 iniţială;
- $Z_o \in \Gamma$ simbolul de start al memoriei stivă;
- $F \subseteq Q$ mulţimea stărilor finale;
- $\delta : Q x (\Sigma \cup \{\varepsilon\}) x \Gamma \rightarrow \mathcal{P}(Q x \Gamma^*)$ funcţia de tranziţie
  - are ca valori submulţimi finite din $Q x \Gamma^*$ (posibil multimea vida)

Determinism

M = (Q, Σ, Γ, δ, q_o, Z_o, F) este ***determinist*** dacă:

$\forall$**Z**$\in$**Γ**, $\forall$**q**$\in$**Q** , $\forall$**a**$\in$Σ

1) **|δ(q,ε,Z)| = 0** si **|δ(q,a,Z)| <=1**

2) **|δ(q,ε,Z)| = 1** si **|δ(q,a,Z)| = 0**

In caz contrar, automatul nu este determinist

Multimea limbajelor acceptate de APD nedeterministe este strict mai larga decat multimea limbajelor acceptate de APD deterministe

exemplu

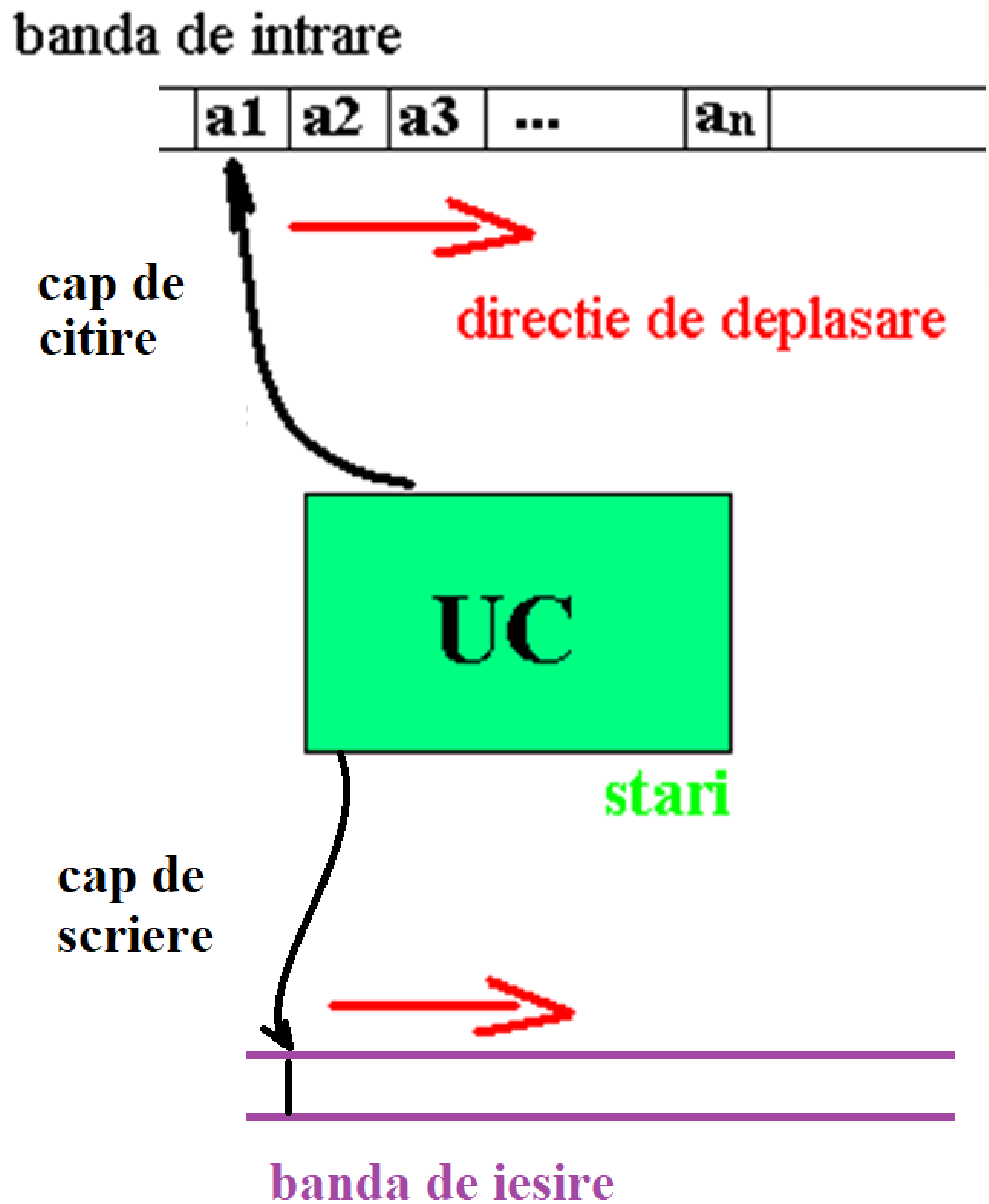
Translator finit

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

- Q alfabetul stărilor;
- Σ alfabetul de intrare;
- D alfabetul de iesire;
- $q_o \in Q$ stare iniţială;
- $F \subseteq Q$ mulţimea stărilor finale;
- $\delta : Q x(\Sigma \cup \{\varepsilon\}) \to \mathcal{P}_0 (Q \text{ x } D^*)$

multimea partilor finite

Translator finit



Translator finit

Exemplu:

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

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

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

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

Translatarea definita de M:

$T(M)=\{(x,y) | \ x\in\Sigma^* , y\in D^*, (q_0,x,\varepsilon) \ |\!\!-^* (q,\varepsilon,y), q\in F)$

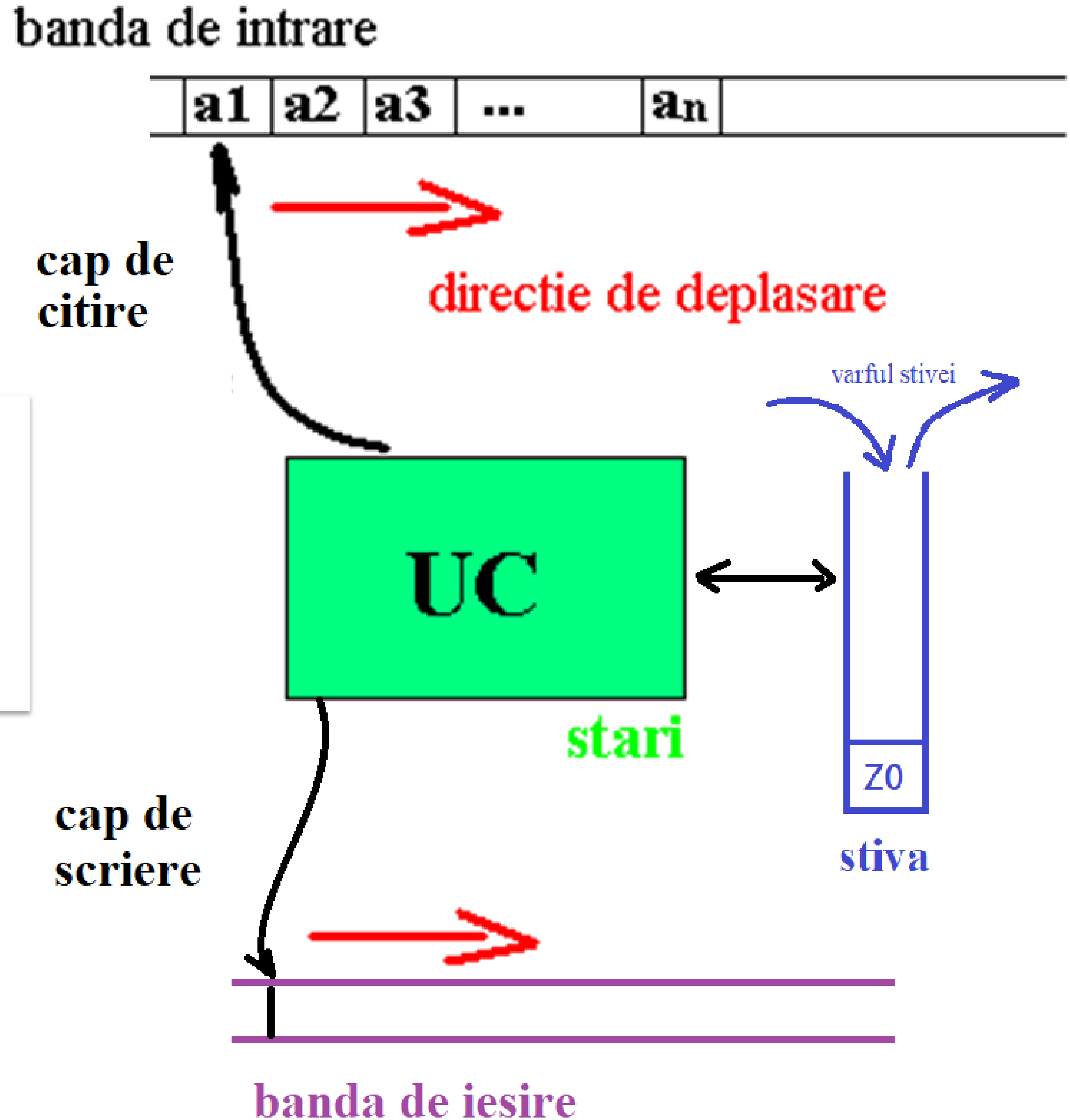
Translator push-down

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

- Q alfabetul stărilor;
- Σ alfabetul de intrare;
- Γ alfabetul memoriei stivă;
- D alfabetul de iesire;
- $q_o \in Q$ stare iniţială;
- $Z_o \in \Gamma$ simbolul de start al memoriei stivă;
- $F \subseteq Q$ mulţimea 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



Translator push-down

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

$Q = \{q\}$

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

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

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

$q_o=q$

$Z_o=E$

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

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

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

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

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

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

… am lucrat si cu alte translatoare

Vezi:

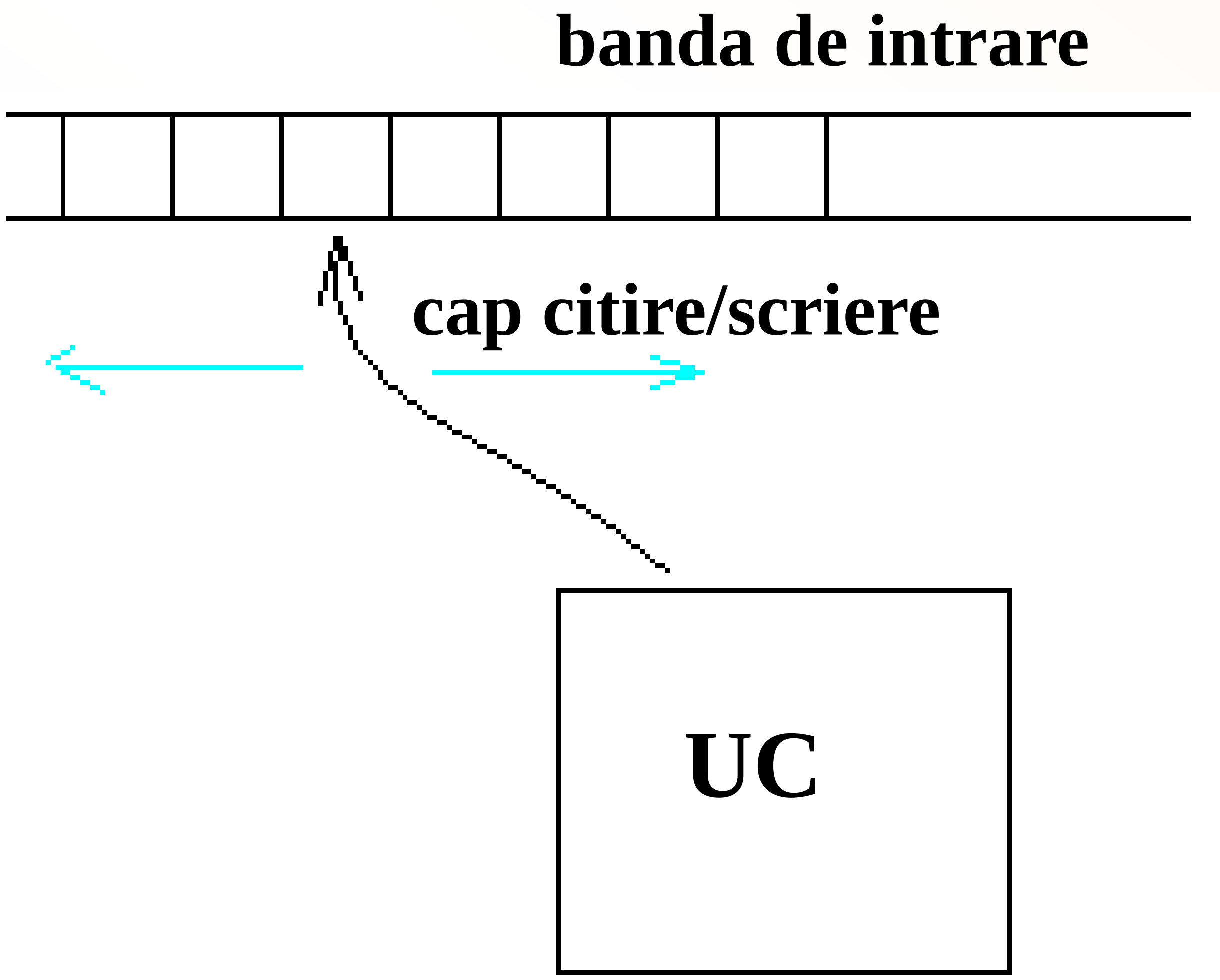
LL(1)

LR(*)

Ne reamintim: Analizorul LL(1)

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

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

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

- tranzitie

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

$$X_1X_2\ldots X_{i-1}qX_i\ X_{i+1}\ldots X_n \vdash X_1X_2\ \ldots\ X_{i-2}\ p\ X_{i-1}YX_{i+1}\ldots X_n$$

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

$$X_1X_2\ldots X_{i-1}qX_i\ X_{i+1}\ldots X_n \vdash X_1X_2\ \ldots\ X_{i-1}Y\ p\ X_{i+1}\ldots X_n$$

- limbaj acceptat

$$\{w \in \Sigma^* \mid q_0w \vdash \alpha_1q\alpha_2\ ,\ q \in F,\ \alpha_1\alpha_2 \in \Gamma^*\}$$

Exemplu: masina Turing

Functia de tranzitie

0011 ?

	Σ		$\Gamma - \Sigma$			
	0	**1**	**X**	**Y**	**#**	
q_0	(q_1,X,R)			(q_3,Y,R)		0
q_1	$(q_1,0,R)$	(q_2,Y,L)		(q_1,Y,R)		0
q_2	$(q_2,0,L)$		(q_0,X,R)	(q_2,Y,L)		0
q_3				(q_3,Y,R)	$(q_4,\#\ ,R)$	0
q_4						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 maşina Turing este modelul cel mai general de calcul care poate fi propus.
  - maşina 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 aceleaşi lucruri
- Acest enunţ nu este demonstrabil în sens matematic.

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

- **Timpul** de calcul pentru un şir dat la intrare: este numărul de mutări făcut de maşina Turing înainte de a intra în starea ``terminat'';
- **Spaţiul** consumat pentru un şir de intrare: este numărul de căsuţe de pe bandă pe care algoritmul le foloseşte în timpul execuţiei sale.