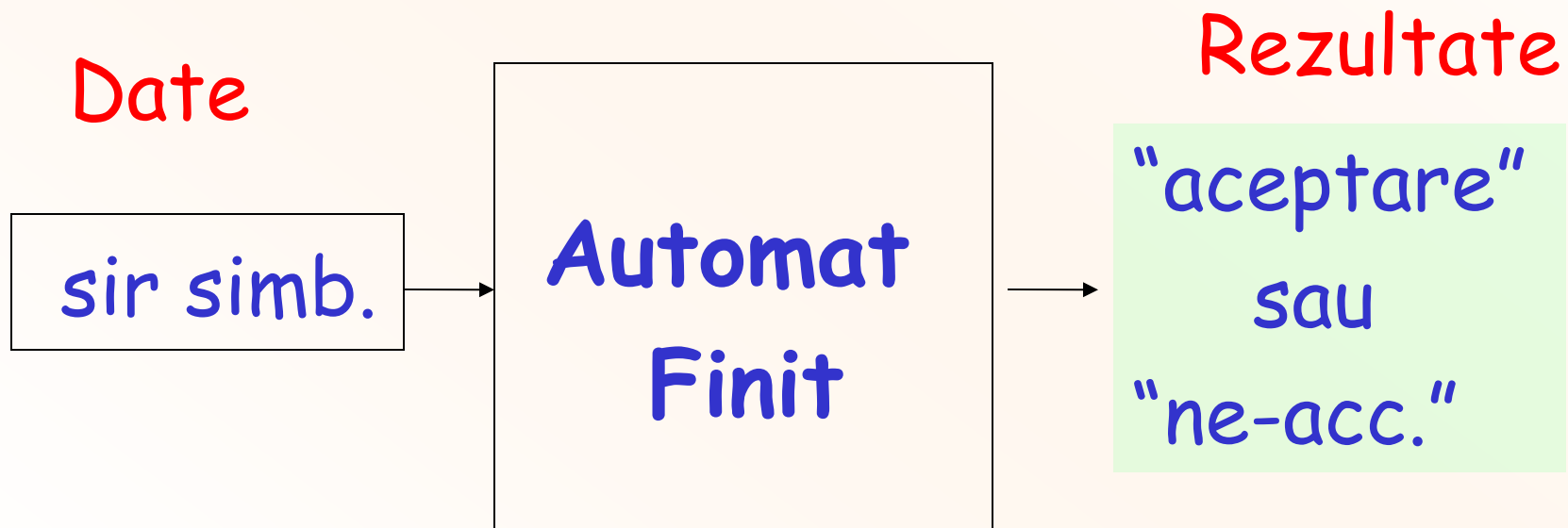


Automat finit (AF)



Automat finit: model fizic

banda de intrare



cap
citire

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

AF – reprezentare tabelara

	Σ			
Q	δ		a_j	
	q_i			z_i

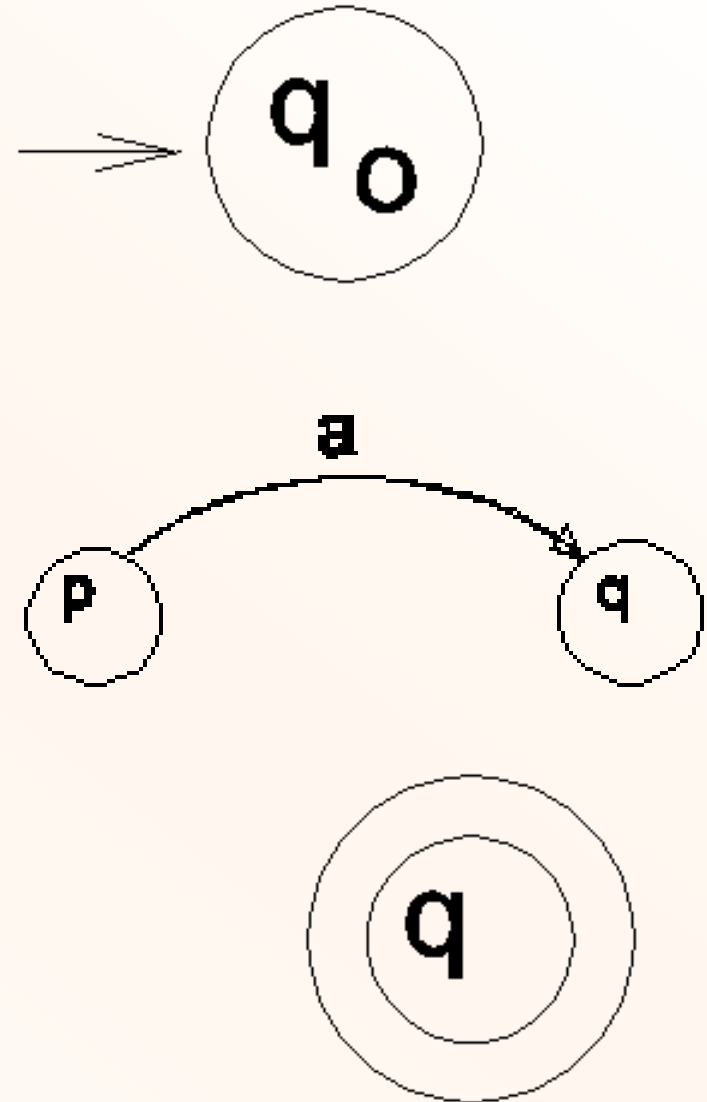
$z_i =$ 0 daca q_i nu e stare finala
 1 daca q_i este stare finala

AF reprezentat tabelar; exemplu

δ	0	1	
p	q	p	0
q	r	p	0
r	r	r	1

AF – reprezentare sub forma de graf

- graf orientat
- cu noduri si arce etichetate
- (graf de tranzitii)



Configuratii si relatii de tranzitie

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

configuratie: $(q, x) \in Q \times \Sigma^*$

tranzitie: element din $(Q \times \Sigma^*) \times (Q \times \Sigma^*)$

- \vdash tranzitie directa
 - \vdash^k k-tranzitie
 - \vdash^+ +-tranzitie
 - \vdash^* *-tranzitie
- $$(p, aw) \vdash (q, w) \iff \delta(p, a) \ni q;$$
- $$p, q \in Q, a \in \Sigma, w \in \Sigma^*$$

Limбай acceptat; autom. echivalente

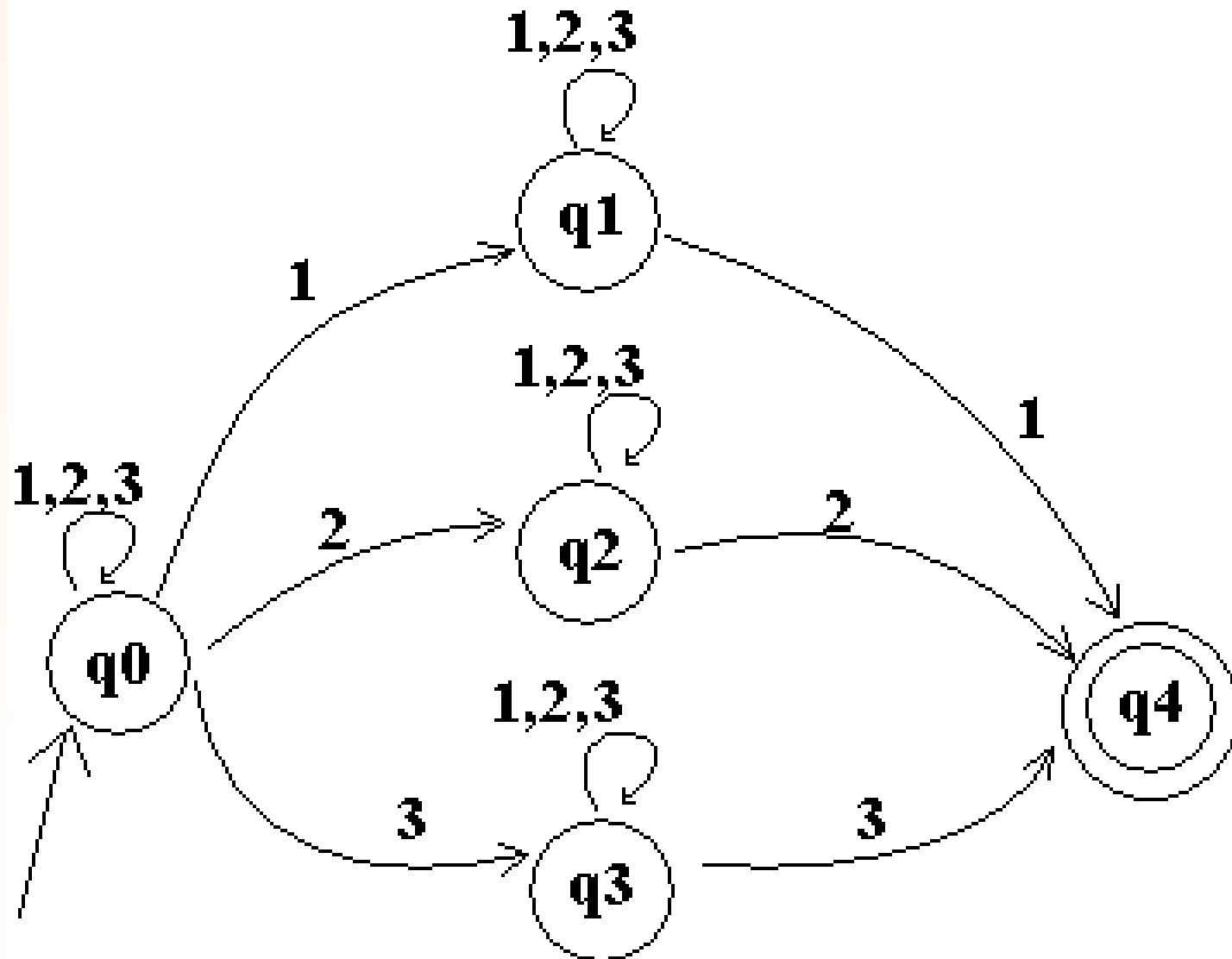
- Limбай acceptat de automat

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

- Automate echivalente

M_1 echivalent cu M_2 daca: $L(M_1) = L(M_2)$

Automat finit - exemplu



Determinism

- Automat finit determinist (AFD)

$$|\delta(q,a)| \leq 1 \quad \forall q \in Q, a \in \Sigma$$

- Automat finit nedeterminist (AFN)

$$\exists q \in Q, a \in \Sigma \text{ astfel incat } |\delta(q,a)| > 1$$

- Automat finit determinist complet definit

$$|\delta(q,a)| = 1 \quad \forall q \in Q, a \in \Sigma$$

Echivalenta dintre AFD si AFN

Teorema:

- $\forall M_1 - \text{AFN} \quad \exists M_2 - \text{AFD echivalent}$

Constructie (nu demonstratie!):

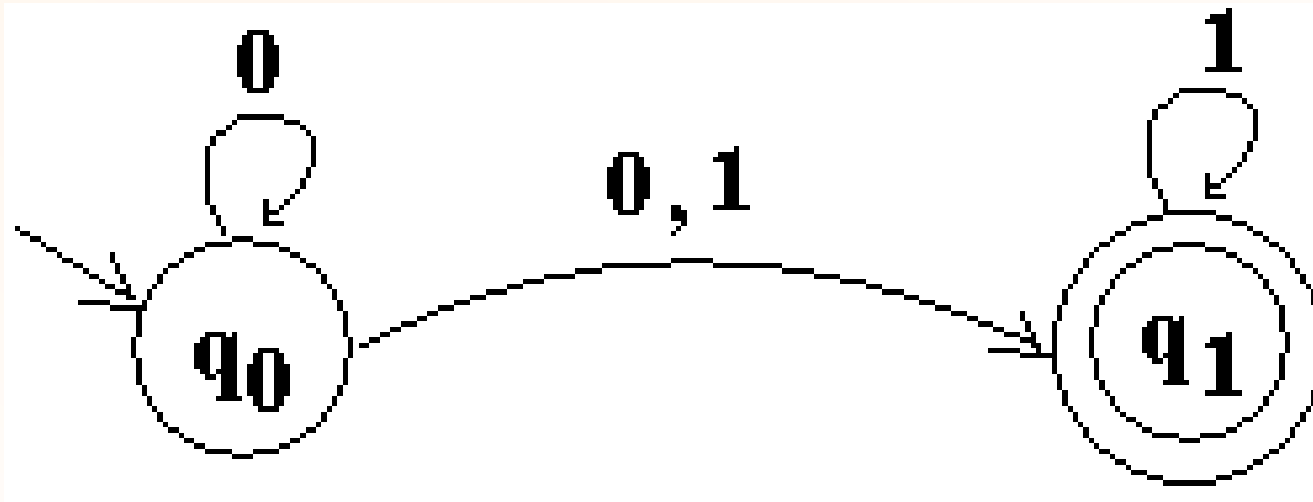
- Pornim cu: $M_1 = (Q_1, \Sigma_1, \delta_1, q_{01}, F_1) - \text{AFN}$ oarecare
- Construim: $M_2 = (Q_2, \Sigma_2, \delta_2, q_{02}, F_2) - \text{AFD}$
pe baza lui M_1
a.i. $L(M_1) = L(M_2)$

Teor: $\forall M_1 - \text{AFN} \quad \exists M_2 - \text{AFD}$ equivalent

- $\Sigma_2 = \Sigma_1$
- $Q_2 = \mathcal{P}(Q_1)$
- $q_{02} = \{q_{01}\}$
- $F_2 = \{S \in \mathcal{P}(Q_1) \mid S \cap F_1 \neq \emptyset\}$
- $\delta_2(q, a) = \{r \in Q_1 \mid \exists q_1 \in q \text{ a.i. } r \in \delta_1(q_1, a)\}$
$$= \bigcup_{q_1 \in q} \delta(q_1, a)$$

M_2 – determinist (?)

Problema: determinati AFD echiv. pt.



Am revenit si in sapt. 14
si am continuat prezentarea
urmatoarelor slide-uri

AF – stări care nu contribuie la acceptarea unui cuvânt

- stare neproductivă – (nu e stare productivă)
- stare inaccesibilă – (nu e stare accesibilă)
- stare productivă: $q \in Q$ a.i.
$$\exists w \in \Sigma^* \text{ si } q_f \in F \text{ a.i. } (q, w) \vdash^*(q_f, \varepsilon)$$
- stare accesibilă: $q \in Q$ a.i.
$$\exists w \in \Sigma^* \text{ a.i. } (q_0, w) \vdash^*(q, \varepsilon)$$

Algoritm determin. stari accesibile

1. $i := 0$

$A_0 := \{q_0\}$

2. **Repeta**

$i := i + 1$

$A_{i+1} := A_i \cup \{ q \in Q \mid \exists p \in A_i, \exists a \in \Sigma \text{ a.i. } q \in \delta(p, a) \}$

pana cand $A_i = A_{i+1}$

$\{ A_i - \text{multimea starilor accesibile} \}$

Algoritm determin. stari productive

1. $i:=0$

$A_0:=F$

2. **Repeta**

$i:=i+1$

$A_{i+1}=A_i \cup \{ q \in Q \mid \exists p \in A_i, \exists a \in \Sigma \text{ a.i. } p \in \delta(q,a) \}$

pana cand $A_i=A_{i+1}$

$\{ A_i - \text{multimea starilor productive} \}$

Teorema:

$\forall M_1 - \text{AF}$ exista $M_2 - \text{AF}$ fara st. neproductive **echiv.**

Constructie (nu demonstratie!):

- Pornim cu: $M_1 = (Q_1, \Sigma_1, \delta_1, q_{01}, F_1) - \text{AF}$ oarecare
- determinam A – multimea starilor productive (algoritmul anterior)
- Construim: M_2 pe baza lui M_1 (a.i. $L(M_1) = L(M_2)$)

$$M_2 = (A, \Sigma_1, \delta_{1/A}, q_{01}, F_1)$$

$$L(M_1) = L(M_2) \quad !$$

Teorema:

$\forall M_1 - \text{AF}$ exista $M_2 - \text{AF}$ fara st. inaccesibile **echiv.**

Constructie (nu demonstratie!):

- ... analog ...

? alta metoda de determinare a AFD echivalent pentru un AFN dat

- $M_1 \Rightarrow ? M_2$

Ideea:

1. $\{q_{01}\} \in Q_2$
pornim cu $Q_2 = \{q_{01}\}$
2. adaugam la Q_2 toate submultimile lui Q_1 la care se poate ajunge prin functia de tranzitie, atunci cand se aplica unei stari $q \in Q_2$ deja adaugata

AFD complet definit

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

- $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ functie de tranzitie ; $|\delta(q,a)| \leq 1$
- ...

Teor: \forall AFD \exists AFD complet definit echivalent

Constructie:

AFD \Rightarrow AFD complet definit:

- adaugam o stare (neproductiva) r si extindem δ astfel:
- $\forall (q,a) \in Q \times \Sigma$ a.i. $\delta(q,a) = \emptyset$ devine: $\delta(q,a) = \{r\}$
- $\forall a \in \Sigma$ $\delta(r,a) = \{r\}$

Minimizarea automatelor finite

Ce vrem:

Automat determinist cu numar minim de stari !

Automat redus

- AFD
- nu contine stari inaccesibile si neproductive
- nu contine perechi de **stari echivalente**

Minimizarea AFD

- automat cu numar minim de stari
 - fara stari –inaccesibile, neproductive
 - mai putine stari ?

ideea: relatie de echivalenta; clase de echivalenta

- stari diferite
- stari k diferite
- stari echivalente
- stari k -echivalente

Automatul redus

Fie M_1 – un automat finit oarecare

- Determinam AFD echivalent
- Eliminam starile inaccesibile si neproductive
- Determinam AFD echivalent complet definit

Fie $M = (Q, \Sigma, \delta, q_0, F)$ automatul rezultat.

- Determinam relatia \equiv (stari echivalente, clase de echivalenta)
- Pe baza relatiei \equiv determinam automatul:

$$M_{\equiv} = (Q/\equiv, \Sigma, \delta_{\equiv}, [q_0], F_{\equiv})$$

Q/\equiv - multimea claselor de echivalenta

$$\delta_{\equiv}([q], a) = [\delta(q, a)]$$

$$F_{\equiv} = \{[q] \mid q \in F\}$$

Automat redus

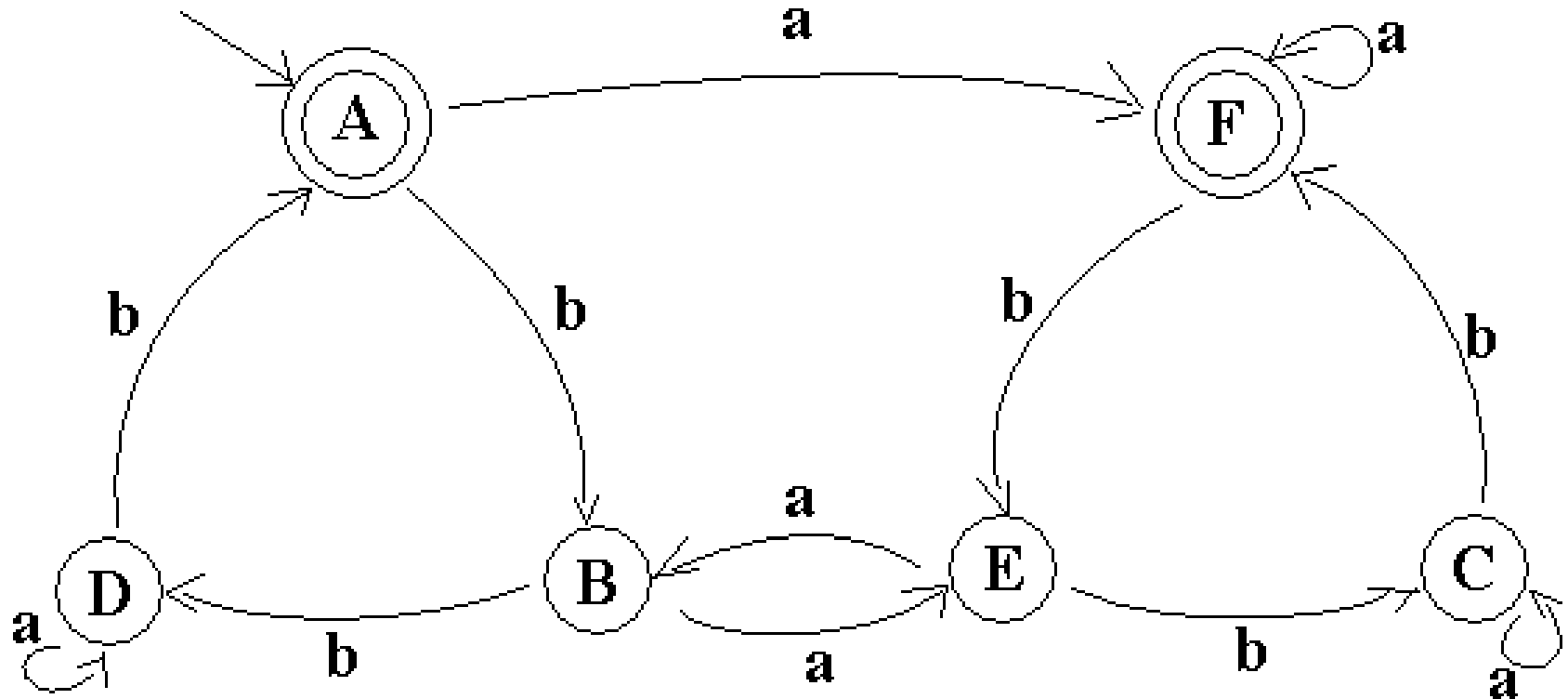
- **Teorema**

Automatul redus are numar minim de stari
dintre toate AFD echivalente

- **Teorema**

$\forall M_1 - AF \exists M_2 - \text{automat redus echivalent}$

Determinati clasele de echivalenta ale starilor automatului de mai jos



determinati automatul redus !

Stari diferite

- o alta exprimare

q_1, q_2 sunt *stari diferite* de $x \in \Sigma^*$

daca $\exists q_f \in F$ a.i. $(q_1, x) \vdash^*(q_f, \varepsilon)$

si nu exista nici un $q \in F$ a.i. $(q_2, x) \vdash^*(q, \varepsilon)$

sau

daca $\exists q_f \in F$ a.i. $(q_2, x) \vdash^*(q_f, \varepsilon)$

si nu exista nici un $q \in F$ a.i. $(q_1, x) \vdash^*(q, \varepsilon)$

- x (de mai sus) *diferentiaza* pe q_1 si q_2

Stari diferite

- o alta exprimare

PP. AFD complet definit

q_1, q_2 sunt *stari diferite* de $x \in \Sigma^*$

daca $\exists r_1, r_2$ astfel incat: $(q_1, x) \vdash^*(r_1, \varepsilon)$

si $(q_2, x) \vdash^*(r_2, \varepsilon)$

are loc una dintre:

1. $r_1 \in F$ si $r_2 \in Q-F$
2. $r_1 \in Q-F$ si $r_2 \in F$

Relatii intre stari

- q_1, q_2 - stari diferite
– cf. def. de mai sus
– ($\exists x \in \Sigma^*$ care sa le diferentieze)
- stari k diferite
– daca $\exists x \in \Sigma^*, |x| \leq k$ care sa le diferentieze
- stari echivalente (\equiv)
– daca nu exista $x \in \Sigma^*$ care sa le diferentieze
- stari k-echivalente (\equiv^k)
– daca nu exista $x \in \Sigma^*, |x| \leq k$, care sa le diferentieze

Proprietati ale rel. de k-echivalenta (\equiv^k)

- $q_1 \equiv^0 q_2$ ddaca $(q_1, q_2 \in F)$ sau $(q_1, q_2 \in Q-F)$
- $\equiv^0 \supseteq \equiv^1 \supseteq \equiv^2 \supseteq \dots \supseteq \equiv^n \supseteq \dots$
- Daca $(\equiv^k) = (\equiv^{k+1})$ atunci $(\equiv^k) = \equiv$

Lema:

Pt. orice M exista $n \in \mathbf{N}$ a.i. $q_1 \equiv^n q_2 \Rightarrow q_1 \equiv q_2$

- **ideea:** pot avea un nr. finit de relatii distincte (max. $|Q|$)