Curs 5

II.2.3 Acţiuni ale monoizilor

Observaţia II.2.3.1. *Fie $(M, \cdot, e)$ un monoid. Atunci M, împreună cu operaţia binară $m \cdot_{op} n = n \cdot m, \forall m, n \in M$, formează un monoid cu acelaşi element neutru e, numit* monoidul opus *lui M şi notat M^{op}. Dacă M este comutativ, atunci clar M^{op} coincide cu M. Dacă M este grup, atunci funcţia $M \longrightarrow M^{op}$, $m \mapsto m^{-1}$ este un izomorfism de monoizi. Dacă M nu este grup şi nici comutativ, atunci monoizii M şi M^{op} nu mai sunt neapărat izomorfi.*

Exemplul II.2.3.2. *Fie $M = \{e, a, b\}$ cu operaţia binară $a \cdot a = a \cdot b = a$, $b \cdot a = b \cdot b = b$ şi elementul neutru e. Se verifică uşor că M este un monoid necomutativ. Dacă ar exista un izomorfism $f : M \longrightarrow M^{op}$, atunci f trebuie să fie bijectivă şi $f(e) = e$. Există numai două cazuri posibile: $f = \mathbf{1}_M$ sau*

$$f(a) = b, \quad f(b) = a$$

Dar $\mathbf{1}_M : M \longrightarrow M^{op}$ nu e morfism de monoizi. Rămâne deci al doilea caz. Din condiţia $f(a \cdot b) = f(a) \cdot_{op} f(b) = f(b) \cdot f(a)$, rezultă $b = f(a) = f(a \cdot b) = f(b) \cdot f(a) = a \cdot b = a$, fals.

Definiţia II.2.3.3. *Fie $(M, \cdot, e)$ un monoid şi X o mulţime. Spunem că M* acţionează *pe X la dreapta (sau că X este o M-mulţime) dacă există o funcţie $\delta : X \times M \longrightarrow X$ astfel încât:*

(A1) $\delta(x, e) = x, \forall x \in X$.

(A2) $\delta(x, m \cdot n) = \delta(\delta(x, m), n), \forall x \in X,\ m, n \in M$.

*Vom nota $\delta(x, m) = x \lhd m$; spunem că elementul $m \in M$ acţionează pe $x \in X$ prin $x \lhd m \in X$. Relaţiile **(A1)** - **(A2)** se scriu atunci:*

$$x \lhd e = x \text{ şi } x \lhd (m \cdot n) = (x \lhd m) \lhd n$$

Exemplul II.2.3.4. *(i) Fie $(M, \cdot, e)$ un monoid şi X o mulţime. Atunci funcţia $\delta : X \times M \longrightarrow X, \delta(x, m) = x$ (deci scriem $x \lhd m = x$), defineşte o acţiune a lui M pe X, numită acţiune* trivială*: prin construcţie, avem $x \lhd e = x$ şi $(x \lhd m) \lhd n = x \lhd n = x = x \lhd (m \cdot n)$, pentru orice $x \in X, m, n \in M$.*

(ii) Fie $(M, \cdot, e)$ un monoid. Atunci operaţia binară "$\cdot$" pe M determină o acţiune a monoidului pe el însuşi $\delta : M \times M \longrightarrow M, \delta(x, m) = x \cdot m, \forall x, m \in M$, numită acţiune regulată *(deci scriem $x \lhd m = x \cdot m$). Axiomele **(A1)**-**(A2)** sunt verificate: $x \lhd e = x \cdot e = x$ şi $(x \lhd m) \lhd n = (x \cdot m) \cdot n = x \cdot (m \cdot n) = x \lhd (m \cdot n)$.*

(iii) Fie $(M, \cdot, e)$ monoid şi X o mulţime pe care M acţionează la dreapta prin $X \times M \longrightarrow X$, $(x, m) \mapsto x \lhd m$. Atunci funcţia $\mathcal{P}(X) \times M \longrightarrow \mathcal{P}(X)$, $(A, m) \mapsto A \lhd m = \{x \lhd m \mid x \in A\}, \forall A \subseteq X, m \in M$, defineşte o acţiune a lui M pe mulţimea părţilor lui X. Verificăm:

$$\begin{aligned} A \lhd e &= \{x \lhd e \mid x \in A\} \\ &= \{x \mid x \in A\} \\ &= A \\ (A \lhd m) \lhd n &= \{x \lhd m \mid x \in A\} \lhd n \\ &= \{(x \lhd m) \lhd n \mid x \in A\} \\ &= \{x \lhd (m \cdot n) \mid x \in A\} \\ &= A \lhd (m \cdot n) \end{aligned}$$

Exerciţiul II.2.3.5. *Dacă $(M, \cdot, e)$ este un monoid şi X o mulţime pe care M acţionează la drepta, atunci M acţionează şi pe produsul cartezian $X^n = \underbrace{X \times \ldots \times X}_{n \text{ ori}}, \forall n \in \mathbb{N}^*$.*

II.2.4 Submonoizi. Morfisme de monoizi. Monoid factor

Definiţia II.2.4.1. *Fie $(M, \cdot, e)$ un monoid şi $N \subseteq M$. Spunem că N este un* submonoid *dacă:*

(S1) $e \in N$.

(S2) $\forall m, n \in N \Longrightarrow m \cdot n \in N$.

Orice submonoid este în particular un monoid cu operaţia binară indusă, dar nu orice submulţime a unui monoid care este la rândul său monoid cu operaţia

indusă este submonoid. De exemplu, fie $M = (\{a,b\}, \circ, \mathbf{1}_{\{a,b\}})$ monoidul tuturor funcţiilor $\{a,b\} \longrightarrow \{a,b\}$ cu operaţia de compunere. Luăm $N = \{f_a\}$, unde f_a este funcţia constantă în a. Atunci $f_a \circ f_a = f_a$, deci N este monoid în raport cu compunerea, dar N nu e submonoid deoarece $\mathbf{1}_{\{a,b\}} \notin N$.

Observaţia II.2.4.2. *Fie $N \subseteq M$ submonoid şi X o mulţime pe care M acţionează la dreapta prin $\delta : X \times M \longrightarrow X$. Atunci restricţia lui δ la N, $\delta|_{X\times N} : X \times N \longrightarrow X$ defineşte o acţiune a lui N pe X.*

Fie $(M, \cdot, e)$ un monoid şi $X, Y \subseteq M$. *Produsul* submulţimilor X, Y este submulţimea

$$X \cdot Y = \{x \cdot y \mid x \in X, y \in Y\}$$

Se obţine astfel o operaţie binară pe mulţimea submulţimilor lui M $\mathcal{P}(M)$; $\mathcal{P}(M)$, împreună cu această operaţie formează un monoid cu elementul neutru $\{e\}$ şi element zero $\emptyset$. De remarcat că dacă $N \subseteq M$ este submonoid, atunci N este element idempotent (adică $N \cdot N = N$) în $\mathcal{P}(M)$ (exerciţiu!).

Reamintim: fie A o mulţime, pe care o vom numi *alfabet*. Un *string* α (de lungime n) este un element din $A^n = \underbrace{A \times \cdots \times A}_{\text{n ori}}, \forall n \in \mathbb{N}^*$, scris $\alpha = a_1 \ldots a_2$, fără paranteze sau virgulă. Convenim existenţa unui string vid (de lungime 0), notat ϵ. Mulţimea tuturor stringurilor se va nota A^* (formal, $A^* = \cup A^n, n \geq 0$, unde $A^0 = \{\epsilon\}$). Pe A^*, definim o operaţie numită *concatenare* prin:

$$\alpha \cdot \beta = a_1 \ldots a_n b_1 \ldots b_m$$

dacă $\alpha = a_1 \ldots a_n \in A^*$, $\beta = b_1 \ldots b_m \in A^*$. De asemenea, convenim că

$$\alpha \cdot \epsilon = \epsilon \cdot \alpha, \forall \alpha \in A^*$$

Atunci $(A^*, \cdot, \epsilon)$ devine un monoid, numit *monoidul liber* peste A.

Definiţia II.2.4.3. *Fie $(M, \cdot, e_M), (N, \cdot, e_N)$ monoizi. O funcţie $f : M \longrightarrow N$ se numeşte* morfism de monoizi *dacă:*

(M1) $f(e_M) = f(e_N)$

(M2) $f(mm') = f(m)f(m'), \forall m, m' \in M$

Un izomorfism de monoizi *este un morfism bijectiv.*

Exemplul II.2.4.4. *Funcţia $l : A^* \longrightarrow \mathbb{N}, l(\alpha) = |\alpha|$ este un morfism de monoizi, căci $|\epsilon| = 0$ şi $|\alpha\beta| = |\alpha| + |\beta|$, $\forall \alpha, \beta \in A^*$. Dacă $A = \{a\}$, atunci l este chiar un izomorfism.*

Observaţia II.2.4.5. *Fie $f : M \longrightarrow N$ un morfism de monoizi. Atunci pentru orice $M' \subseteq M$ submonoid, $f(M') \subseteq N$ este submonoid. Reciproc, dacă $N' \subseteq N$ e submonoid, atunci $f^{-1}(N') \subseteq M$ e submonoid. În particular, dacă luăm $M' = M$, obţinem că $\mathsf{Im} f$ e submonoid în N.*

Demonstraţie. Reamintim că $f(M') = \{f(m) \mid m \in M'\}$ şi $f^{-1}(N') = \{m \in M \mid f(m) \in N'\}$. Arătăm acum că $f(M')$ e submonoid: cum $e_N = f(e_M)$ şi $e_M \in M'$ (M' este submonoid), rezultă $e_N \in f(M')$. Acum ,fie $m, m' \in M'$. Atunci $f(m)f(m') = f(m \cdot m') \in f(M')$ deoarece $m \cdot m' \in M'$. Trecem la a doua afirmaţie, privind $f^{-1}(N')$: $f(e_M) = e_N \in N'$, deci $e_M \in f'(N')$. Dacă $m, m' \in f'(N')$, atunci $f(m) \cdot f(m') \in N'$. Dar N' e submonoid, deci $f(m \cdot m') = f(m) \cdot f(m') \in N'$, adică $m \cdot m' \in N'$. □

Definiţia II.2.4.6 (Congruenţă)**.** *O relaţie de echivalenţă $\mathcal{R}$ pe monoidul M se numeşte* congruenţă *dacă*

(C) $m\mathcal{R}m' \Longrightarrow (m \cdot n)\mathcal{R}(m' \cdot n)$ *şi* $(n \cdot m)\mathcal{R}(n \cdot m'), \forall m, m', n \in M$.

Reamintim că fiecărei relaţii de echivalenţă $\mathcal{R}$ îi corespunde o partiţie a monoidului M în clase de echivalenţă $([m]_{\mathcal{R}})_{m \in M}$, unde $[m]_{\mathcal{R}} = \{n \in M \mid m\mathcal{R}n\}$. Mulţimea claselor de echivalenţă s-a notat cu $M/\mathcal{R}$ şi se numeşte mulţimea factor.

Exemplul II.2.4.7. *Pe A^* luăm relaţia*

$$\alpha \mathcal{R} \beta \Longleftrightarrow \mid \alpha \mid = \mid \beta \mid$$

Este în mod evident o relaţie de echivalenţă; în plus, pentru orice $\gamma \in A^*$, $\mid \alpha\gamma \mid = \mid \alpha \mid + \mid \gamma \mid = \mid \beta \mid + \mid \gamma \mid = \mid \beta\gamma \mid$ şi analog $\mid \gamma\alpha \mid = \mid \gamma\beta \mid$, deci $\mathcal{R}$ este o congruenţă.
Dacă $\mid \alpha \mid = n$, atunci $[\alpha]_{\mathcal{R}} = \{\beta \in A^* \mid \alpha\mathcal{R}\beta\} = A^n$ este mulţimea tuturor stringurilor de lungime n.

Dacă $(M, \cdot, e)$ este un monoid şi $\mathcal{R}$ o congruenţă pe M, atunci putem defini o operaţie binară pe mulţimea factor $M/\mathcal{R}$ prin:

$$[m]_{\mathcal{R}} \cdot [m']_{\mathcal{R}} = [m \cdot m']_{\mathcal{R}}, \forall m, m' \in M$$

Această definiţie nu depinde de reprezentanţii claselor de echivalenţă: dacă $[m]_{\mathcal{R}} = [n]_{\mathcal{R}}$ şi $[m']_{\mathcal{R}} = [n']_{\mathcal{R}}$, atunci $m\mathcal{R}n$ şi $m'\mathcal{R}n'$. Rezultă $m \cdot m'\mathcal{R}n \cdot m'$ şi $n \cdot m'\mathcal{R}n \cdot n'$, de unde $m \cdot m'\mathcal{R}n \cdot n'$, adică $[m \cdot m']_{\mathcal{R}} = [n \cdot n']_{\mathcal{R}}$. Mai mult, operaţia astfel definită este asociativă şi admite elementul neutru $[e]_{\mathcal{R}}$ (exerciţiu!), deci $(M/\mathcal{R}, \cdot, [e]_{\mathcal{R}})$ este un monoid. Fie $\pi_{\mathcal{R}} : M \longrightarrow M/\mathcal{R}$ funcţia care asociază fiecărui element clasa sa de echivalenţă: $\pi_{\mathcal{R}}(m) = [m]_{\mathcal{R}}$ ($\pi_{\mathcal{R}}$ se numeşte *surjecţie canonică*). Atunci $\pi_{\mathcal{R}}$ este surjectivă (de ce?) şi morfism de monoizi: $\pi_{\mathcal{R}}(e) = [e]_{\mathcal{R}}$ prin construcţie şi $\pi_{\mathcal{R}}(m \cdot m') = [m \cdot m']_{\mathcal{R}} = [m]_{\mathcal{R}}[m']_{\mathcal{R}} = \pi_{\mathcal{R}}(m)\pi_{\mathcal{R}}(m')$, $\forall m, m' \in M$.

Congruenţă nucleară. Fie $f : M \longrightarrow N$ o funcţie. Considerăm relaţia $m\mathcal{R}_f m' \iff f(m) = f(m')$, $\forall m, m' \in M$. Atunci $\mathcal{R}_f$ este o relaţie de echivalenţă, ale cărei clase vor fi $(f^{-1}(n))_{n\in \mathsf{Im}f}$. Dacă M, N sunt monoizi şi f este morfism, atunci $\mathcal{R}_f$ este chiar o congruenţă: fie $m, m' \in M$ astfel încât $m\mathcal{R}_f m'$. Atunci $f(m) = f(m')$. Dacă $n \in M$, avem $f(m\cdot n) = f(m)\cdot f(n) = f(m')\cdot f(n) = f(m'\cdot n)$, deci $(m\cdot n)\mathcal{R}(m'\cdot n)$ şi analog la stânga.
Să remarcăm că dacă $\mathcal{R}$ este o congruenţă atunci congruenţa asociată surjecţiei canonice $\pi_\mathcal{R}$ este chiar $\mathcal{R}$: dacă $m\mathcal{R}_{\pi_\mathcal{R}} m'$, atunci $\pi_\mathcal{R}(m) = \pi_\mathcal{R}(m')$, deci $[m]_\mathcal{R} = [m']_\mathcal{R}$, echivalent cu $m\mathcal{R}m'$. În consecinţă, a da o congruenţă $\mathcal{R}$ pe un monoid M este acelaşi lucru cu a da un morfism de monoizi cu domeniul M (pornind de la congruenţa $\mathcal{R}$, obţinem morfismul de monoizi $\pi_\mathcal{R}$; invers, dacă $f : M \longrightarrow N$ este un morfism de monoizi, atunci obţinem congruenţa $\mathcal{R}_f$).

Teorema II.2.4.8. *Fie $f : M \longrightarrow N$ un morfism de monoizi. Atunci funcţia $\widetilde{f} : M/\mathcal{R}_f \longrightarrow N$, $\widetilde{f}([m]_{\mathcal{R}_f}) = f(m)$ este un morfism injectiv de monoizi cu imaginea $\mathsf{Im}f$. În particular, rezultă că există un izomorfism de monoizi*

$$M/_{\mathcal{R}_f} \cong \mathsf{Im}f$$

Demonstraţie. Arătăm mai întâi că $\widetilde{f}$ e bine definită şi injectivă: dacă $[m]_{\mathcal{R}_f} = [m']_{\mathcal{R}_f}$, atunci $m\mathcal{R}_f m'$, de unde $f(m) = f(m')$. $\widetilde{f}$ e morfism de monoizi:

$$\widetilde{f}([m]_{M\mathcal{R}_f}) = f(e) = e$$

şi

$$\begin{aligned}\widetilde{f}([m]_{\mathcal{R}_f}\cdot[m']_{\mathcal{R}_f}) &= \widetilde{f}([m\cdot m']_{\mathcal{R}_f})\\ &= f(m\cdot m')\\ &= f(m)\cdot f(m')\\ &= \widetilde{f}([m]_{\mathcal{R}_f})\cdot\widetilde{f}([m']_{\mathcal{R}_f})\end{aligned}$$

Din construcţie, $\mathsf{Im}\widetilde{f} = \mathsf{Im}f$, deci avem izomorfismul $M/\mathcal{R}_f \cong \mathsf{Im}f$. □

Exemplul II.2.4.9. *Fie $A=\{a, b\}$ şi $M = A^*$. Am văzut mai devreme că funcţia $l : A^* \longrightarrow \mathbb{N}$, $l(\alpha) =\mid \alpha \mid$ este un morfism de monoizi (surjectiv). Fie $\mathcal{R}_l$ congruenţa asociată:*

$$\alpha\mathcal{R}_l\beta \Longleftrightarrow \mid \alpha \mid = \mid \beta \mid, \forall\alpha, \beta \in A^*$$

Atunci $A^*/\mathcal{R}_l \cong (N)$ este un izomorfism de monoizi, dat de de $A^* \mapsto n$ (reamintim partiţia în clase de echivalenţă: dacă $\alpha \in A^*$, $\mid \alpha \mid = n$, atunci $[\alpha]_{\mathcal{R}_l} = A^n$), unde operaţia binară pe $A^n/_{\mathcal{R}_l}$ este $(A^n, A^m) \mapsto A^{n+m}$.

Reamintim că pentru orice alfabet A, monoidul A^* are următoarea proprietate : pentru orice monoid M şi pentru orice funcţie $f : A \longrightarrow M$, există un unic morfism de monoizi $f^* : A^* \longrightarrow M$ care extinde pe f (adică $f^*(a) = f(a)$, $\forall a \in A$), şi anume:

$$\begin{aligned} f^*(\epsilon) &= e_M \\ f^*(\alpha) &= f(a_1) \cdot \ldots \cdot f(a_n), \text{ dacă } \alpha = a_1 ... a_n \in A^* \end{aligned}$$

Fie acum X o mulţime şi $\delta : X \times A^* \longrightarrow X$ o acţiune a monoidului stringurilor pe X. Cum $A \subseteq A^*$, prin restricţie se obţine o funcţie $\delta_A : X \times A \longrightarrow X$. Reciproc, orice functie $\delta_A : X \times A \longrightarrow X$ determină o acţiune a monoidului A^* pe X, prin functia $\delta : X \times A \longrightarrow X$, definită recursiv astfel:

$$\begin{aligned} \delta(x, \epsilon) &= x \\ \delta(x, \alpha a) &= \delta_A(\delta(x, \alpha), a), \ \forall \alpha \in A^*, a \in A \end{aligned}$$

Atunci funcţia δ astfel definită verifică în mod clar (A1). Verificăm şi (A2) $\delta(x, \alpha\beta) = \delta(\delta(x, \alpha), \beta)$, prin inducţie după $\mid \beta \mid$: dacă $\mid \beta \mid = 0$, adică $\beta = \epsilon$, atunci

$$\begin{aligned} \delta(\delta(x, \alpha), \beta) &= \delta(\delta(x, \alpha), \epsilon) \\ &= \delta(x, \alpha) \\ &= \delta(x, \alpha\epsilon) \\ &= \delta(x, \alpha\beta) \end{aligned}$$

Presupunem acum afirmaţia adevărată pentru stringuri de lungime mai mică sau egală cu n şi fie $\beta \in A^*$, $\mid \beta \mid = n + 1$. Atunci putem scrie $\beta = \gamma a$, cu $\gamma \in A^*$, $\mid \gamma \mid \leq n$, $a \in A$ şi avem

$$\begin{aligned} \delta(x, \alpha\beta) &= \delta(x, \alpha\gamma a) \\ &= \delta_A(\delta(x, \alpha\gamma), a) \\ &= \delta_A(\delta(\delta(x, \alpha), \gamma), a) \\ &= \delta(\delta(x, \alpha), \gamma a) \\ &= \delta(\delta(x, \alpha), \beta) \end{aligned}$$

II.2.5 Automate

Definiţia II.2.5.1. *Un* automat determinist finit *(DFA) este un tuplu* $\mathcal{A} = (X, A, I, F, \delta)$*, unde*

(i) X *este o mulţime finită, numită* mulţimea stărilor*;*

(ii) A *este un alfabet finit;*

(iii) $I \subseteq X$ *este o submulţime numită* mulţimea stărilor iniţiale*, iar* $F \subseteq X$ mulţimea stărilor finale*;*

(iv) $\delta : X \times A \longrightarrow X$ *este o funcţie numită* funcţia de tranziţie.

Din observaţia precedentă rezultă că un DFA peste un alfabet A este de fapt o mulţime peste care acţioneaza monoidul stringurilor A^* , împreună cu două submulţimi bine determinate (submulţimile stărilor de intrare şi de ieşire).

Definiţia II.2.5.2. *Un automat nedeterminist finit (NDFA) este un tuplu* $\mathcal{A} = (X, A, I, F, E)$ *ca mai sus, cu diferenţa că în loc de o funcţie de tranziţie* $\delta : X \times A \longrightarrow X$*, avem o relaţie* $E \subseteq X \times A \times X$*.*

Dacă $(x, a, y) \in E$ sau $\delta(x, a) = y$, vom nota $x \xrightarrow{a} y$ şi vom spune că automatul a trecut din starea x în starea y la primirea input-ului a.

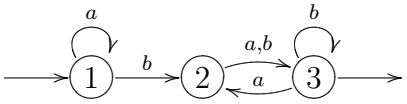
Să remarcăm că o relaţie $E \subseteq X \times A \times X$ este echivalentă cu o funcţie $\tau : X \times A \longrightarrow \mathcal{P}(X)$, unde $\tau(x, a) = \{y \in X \mid (x, a, y) \in E\}$ (şi reciproc, pentru orice functie $\tau : X \times A \longrightarrow \mathcal{P}(X)$, obţinem o relaţie E prin $(x, a, y) \in E \Leftrightarrow y \in \tau(x, a)$). Putem extinde o asemenea funcţie la mulţimea părţilor prin $\mathcal{P}(X) \times A \longrightarrow \mathcal{P}(X)$, $(X', a) \mapsto \{y \in X \mid \exists x \in X', y \in \tau(x, a)\}$, unde $X' \subseteq X$. În particular, fiecărui NDFA $\mathcal{A} = (X, A, I, F, E)$ îi corespunde un DFA prin $\mathsf{Det}(\mathcal{A}) = (\mathcal{P}(X), A, \mathcal{I}, \mathcal{F}, \delta)$, unde $\mathcal{I} = \{I\} \subseteq \mathcal{P}(X)$, $\mathcal{F} = \{X' \subseteq X \mid X' \cap F \neq \emptyset\}$ şi τ ca mai sus. De asemenea, să remarcăm că şi un NDFA corespunde unei acţiuni a monoidului stringurilor A^*, numai că de această dată mulţimea pe care se acţioneaza este $\mathcal{P}(X)$ şi nu X.

În continuare, vom reprezenta automatele (NDFA sau DFA) prin grafuri orientate, astfel: nodurile grafului vor fi stările din X, iar între două noduri x şi y există un arc cu eticheta $a \in A \Longleftrightarrow (x, a, y) \in E$ (sau $\delta(x, a) = y$). Stările finale, respectiv iniţiale vor fi marcate prin săgeţi care ies, respectiv intră din noduri.

Exemplul II.2.5.3. *Fie* $\mathcal{A} = (X = \{1, 2, 3\}, A = \{a, b\}, I = \{1\}, F = \{2\}, \delta)$*, unde* $\delta : X \times A \longrightarrow X$ *e dată in tabelul următor:*

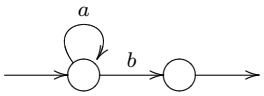
		A	
		a	b
X	1	1	2
	2	3	3
	3	2	3

Atunci automatul va avea reprezentarea

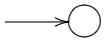


Un drum în automatul $\mathcal{A}$ este o secventă finită de stări consecutive între care există tranziţii, $x_0 \xrightarrow{a_1} x_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} x_n$. Spunem că stringul $\alpha = a_1...a_n$ este asociat drumului. Prin conventie, pentru orice stare $x \in X$ considerăm un drum vid cu stringul asociat ϵ. Spunem că un string $\alpha = a_1 \dots a_n$ este acceptat de automatul $\mathcal{A}$ dacă exista un drum cu $x_0 \in I$ stare iniţială şi $x_n \in F$ stare finală. În particular, stringul vid este acceptat dacă există o stare iniţială care este şi finală. Mulţimea stringurilor acceptate de un automat $\mathcal{A}$ se numeste *limbajul acceptat* de automatul $\mathcal{A}$ şi se va nota cu $L(\mathcal{A})$.

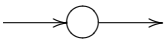
Exemplul II.2.5.4. *(i) Un automat care acceptă limbajul* $\{a^n b \mid n \in \mathbb{N}\}$



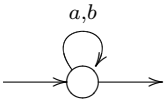
(ii) Un automat care acceptă limbajul vid.



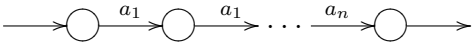
(iii) Un automat care acceptă limbajul format din stringul vid



(iv) Un automat care acceptă toate stringurile peste alfabetul $\{a, b\}$



(v) Un automat care acceptă doar stringul $\alpha = a_1....a_n$



Spunem că două automate sunt *echivalente* dacă acceptă acelaşi limbaj.

Propoziţia II.2.5.5. *Orice NDFA este echivalent cu un DFA.*