

Ex 1

Formal definition of LA :

Fie LA un 6 tuplu $(Q, \Sigma, \Gamma, \delta, q_1, F)$

cu $\delta: \Sigma \times Q \times \Gamma \rightarrow \delta(Q \times \Sigma \times \Gamma)$

(1) • F multime stărilor finale

$$F \subseteq Q$$

(2) • $q_1 \in Q$ stare inițială

(3) • Γ e alfabetul LA (adica al listei din mult. cadrul automatului), finit

(4) • $\Gamma = \{s_1, a_2, a_3\} \cup \{\epsilon\}$ dacă
are loc condițiile:

$$(q_1, a_1, s_1) \rightarrow (q_2, a_2, a_3)$$

$$(q_1, a_1, \epsilon) \rightarrow (q_2, a_2, a_3)$$

$$(q_1, a_1, \epsilon) \rightarrow (q_2, \epsilon, \epsilon)$$

(5) • Σ este multimea alfabetului, multime finită

(6) • Q este multimea stărilor, la fel multime finită

Pentru fiecare simbol s verificat
în cadrul LA-automatului,
avem următoarea funcție de tranziție:

$$\delta(q, a, s) = (m_1, l_1, z_1), s \in LA \quad (1)$$

$$\delta(q, a, s) = (m_2, l_2, z_2), s = \epsilon \quad (2)$$

$$\delta(q, a, s) = (m_3, l_3, z_3), s \notin LA \quad (3)$$

(1) $s \in LA$ (lista), adaug z_1 , scot l_1 (dacă \exists)

(2) $s = \epsilon$, nu verific nimic, pun z_2 în LA,
scot l_1 dacă există

(3) $s \notin LA$, dacă $s \notin LA$, adaug z_3 în LA, scot l_3
dacă există.

2. Simul de caracter s e acceptat dacă automatul LA află într-o stare finală și s e parcurs total. Lista, LA , poate fi goală sau nu.

2. Clear explanation of computation for LA .

Fie $a = a_1 a_2 \dots a_n$ unde fiecare $a_i \in \Sigma$.
Atunci LA acceptă a dacă o secvență de stări r_0, r_1, \dots, r_n în Q există în 3 condiții:

1. $r_0 = q_0$
2. $\delta(r_i, a_{i+1}) = r_{i+1}$ pentru $\forall i = 0, \dots, n-1$
3. $r_n \in F$

Condiția 1 înseamnă că mașina începe în starea de început q_0 .

Condiția 2, înseamnă că mașina merge dintr-o stare în alta în raport cu funcția tranziției.

Condiția 3 înseamnă că simul a e acceptat dacă automatul se termină în stare de accept.

LA recunoaște limbajul A dacă $A = \{w \mid LA \text{ acceptă } w\}$.

Pentru a implementa acest automat, vom condiționa fiecare tranziție de existența unui element.
De exemplu, în cadrul jocului de la ex 3 avem o situație similară.

Fie $\Sigma = \{go, take\}$ acțiunile

$Q = \{kitchen, armoury, entrance, hall, \dots, secret_exit\}$
stările sunt camerele

$\Theta = \{key, invitation, \dots, magic\ wand\}$

inventory - lista din care LA va fi condiționată
Astfel, funcția de tranziție δ va fi condiționată de prezența unui element în inventory.

Ex: pentru a accesa Armoury din Entrance Hall, inventoryul trebuie să conțină 'key' în inventory.