Calculab. cu M. Turing

Def.
Fie $\Sigma_0, \Sigma_1 \not\ni \#$ alfabete, $f: \Sigma_0^* \to \Sigma_1^*$. M. Turing $M = (K, \Sigma, \delta, s)$ calc. f dc. $\Sigma_0, \Sigma_1 \subseteq \Sigma$, $\forall w \in \Sigma_0^*$, $f(w) = u$, atunci $(s, \#w\underline{\#}) \vdash_M^* (h, \#u\underline{\#})$.

Extinderea noțiunii de fcție calc. Turing.

A) $f: \mathbb{N} \to \mathbb{N}$

Fie $I \neq \#$, $\forall n \in \mathbb{N}$, $n \overset{Not}{=} I^n$

o fcție $f: \mathbb{N} \to \mathbb{N}$ este calc. de o M.T. M, dc. M calc. fcția

$$f': \{I\}^* \to \{I\}^*, \text{ unde } f'(I^n) = I^{f(n)}, \ n \in \mathbb{N}.$$

Prin generalizare:

o M.T. calc $f: \mathbb{N}^k \to \mathbb{N}$, dc calc. $f': (\{I\}^*)^k \to \{I\}^*$, unde

$$f'(I^{n_1}, \ldots, I^{n_k}) = I^{f(n_1, \ldots, n_k)}, \ \forall n_1, \ldots, n_k \geq 0.$$

ex:

Fie f fcţia succesor: $f(n) = n+1$, $n \in \mathbb{N}$

$M = (K, \Sigma, \delta, s)$

$K = \{q_0\}$

$\Sigma = \{I, \#\}$

$s = q_0$

δ:

q	σ	$\delta(q, \sigma)$
q_0	I	(h, R)
q_0	$\#$	(q_0, I)

$(q_0, \#II\underline{\#}) \vdash_M (q_0, \#II\underline{I}) \vdash_M (h, \#III\underline{\#})$

în general:

$(q_0, \#I^n\underline{\#}) \vdash_M^* (h, \#I^{n+1}\underline{\#})$

$n=0$, $(q_0, \#\underline{\#}) \vdash_M (q_0, \#\underline{I}) \vdash_M (h, \#I\underline{\#})$

B) Limbaj decidabil Turing

<u>Def.</u>

Fie $\Sigma_0 \not\ni \#$ un alfabet. Atunci limbajul $L \subseteq \Sigma_0^*$ este decidabil Turing dc. și numai dc. funcția $X_L : \Sigma_0^* \to \{Y, N\}$ este calculabilă Turing, unde $\forall w \in \Sigma_0^*$,

$$X_L(w) = \begin{cases} Y, & w \in L \\ N, & w \notin L \end{cases}$$

<u>ex:</u>

Fie $\Sigma_0 = \{a\}$, $L = \{w \in \Sigma_0^* \mid |w| \text{ pară}\}$. $M = (K, \Sigma, \delta, s)$ este o proc. de decizie pt L.

$K = \{q_0, q_1, .., q_6\}$

$\Sigma = \{a, Y, N, \#\}$

$s = q_0$

δ:

q	σ	$\delta(q,\sigma)$
q_0	#	(q_1, L)
q_1	a	$(q_2, \#)$
q_1	#	(q_4, R)
q_2	#	(q_3, L)
q_3	a	$(q_0, \#)$
q_3	#	$(q_6, \#)$
q_4	#	(q_5, Y)
q_5	Y	(h, R)
q_5	N	(h, R)
q_6	#	(q_5, N)

$(q_0, \#a^n\underline{\#}) \vdash_M^* (h, \#Y\underline{\#})$, n este par

$(q_0, \#a^n\underline{\#}) \vdash_M^* (h, \#N\underline{\#})$, n este impar

c) Limbaj Turing acceptat

Def.

Fie $\Sigma_0 \not\ni \#$, alfabet. Spunem că o M. Turing acceptă şirul $w \in \Sigma_0^*$, dacă M.T. se opreşte pt w.

M.T. M acceptă limbajul $L \subseteq \Sigma_0^* \iff L = \{ w \in \Sigma_0^* \mid M \text{ acceptă } w \}$.

ex:

$\Sigma_0 = \{a, b\}$

$L = \{ w \in \Sigma_0^* \mid w \text{ conţine cel puţin 'a'} \}$.

L este acceptat de $M = (K, \Sigma, \delta, s)$,

$K = \{q_0\}$

$\Sigma = \{a, b, \#\}$

$s = q_0$

δ:

q	σ	$\delta(q, \sigma)$
q_0	a	(h, a)
q_0	b	(q_0, L)
q_0	$\#$	(q_0, L)

Compunerea M. Turing

Lema

Fie M o M. Turing și fie $(q_i, w_i \underline{a_i} u_i)$, $i=1,2,3$ config. ale M.T. M.

Dacă $(q_1, w_1 \underline{a_1} u_1) \vdash_M^* (q_2, w w_2 \underline{a_2} u_2)$, pt un șir w

și $(q_2, w_2 \underline{a_2} u_2) \vdash_M^* (q_3, w_3 \underline{a_3} u_3)$

atunci $(q_1, w_1 \underline{a_1} u_1) \vdash_M^* (q_3, w w_3 \underline{a_3} u_3)$.

Ideea dem: dacă M nu se agață în timpul opției lor:

$$(q_2, w_2 \underline{a_2} u_2) \vdash_M^* (q_3, w_3 \underline{a_3} u_3)$$

și cum M nu are o modalitate directă de a detecta capătul stg. al benzii nu trebuie să facă nicio încercare în timpul ac. opții de a-și muta capul la stg. primului simbol din $w_2 a_2 u_2$.

Dar apoi dc. pornește din config. $(q_2, w w_2 \underline{a_2} u_2)$ va ajunge în $(q_3, w w_3 \underline{a_3} u_3)$ fără a pătrunde în primele $|w|$ pătrate ale benzii.

Mașini Turing de bază:

1) scriere simboluri

$|\Sigma|$ mașini de scriere simboluri

$W_a = (K, \Sigma, \delta, s)$

$K = \{q\}$

$s = q$

$\delta(q, b) = (h, a)$, $b \in \Sigma$.

$W_a \stackrel{Not}{=} a$

2) deplas. L/R

$V_L = (\{q\}, \Sigma, \delta_L, q) \stackrel{Not}{=} L$

$\delta_L(q, a) = (h, L)$, $a \in \Sigma$

$V_R = (\{q\}, \Sigma, \delta_R, q) \stackrel{Not}{=} R$

$\delta_R(q, a) = (h, R)$, $a \in \Sigma$

Reguli de compunere a M. Turing

ex: $M_1, M_2 \to$ M. Turing

$$> M_1 \to M_2$$

ex: M_1, M_2, M_3

$$\begin{array}{lll} > M_1 & \xrightarrow{a} & M_3 \\ \downarrow b & & \\ M_2 & & \end{array}$$

Def.

O schemă de M. Turing este un tuplu $(\mathcal{M}, \eta, M_0)$,

$\mathcal{M} \to$ o mulțime finită de M.T. cu alfabetul comun Σ și mulțimi de stări disjuncte

$M_0 \in \mathcal{M} \to$ mașina inițială

$\eta \to$ fcție, $\eta : \mathcal{M} \times \Sigma \to \mathcal{M}$

Fie $\bar{M}$ → M.T. compusă

$\bar{M}$ începe opțiile simulând opțiile M.T. inițiale M_0. Când M_0 se oprește,

$\bar{M}$ începe opțiile altei M.T. din $\mathcal{M}$.

$\eta(M_0, a) \to$ nedef. $\Rightarrow$ $\bar{M}$ se oprește

$\eta(M_0, a) = M \in \mathcal{M}$ $\Rightarrow$ $\bar{M}$ continuă opțiile din st. init. a lui M.

<u>Reguli</u> de reprezentare a unei scheme $(\mathcal{M}, \eta, M_0)$:

1° M.T. M_0 este marcată cu >

2° $a \in \Sigma$, $\eta(M, a) = M'$, săgeată etichetată cu a, $M \xrightarrow{a} M'$

ex: $> R \xrightarrow{a} R$

<u>ex:</u> $\Sigma = \{a, b, c, \#\}$

$$> R \;\substack{\xrightarrow{a} \\ \xrightarrow{b} \\ \xrightarrow{c} \\ \xrightarrow[\#]{}}\; R \quad \overset{\text{Not}}{\equiv} \quad > R \xrightarrow{a, b, c, \#} R \quad \overset{\text{Not}}{\equiv} \quad > R \longrightarrow R \quad \overset{\text{Not}}{\equiv} \quad > RR$$

$$\overset{\text{Not}}{\equiv} \; > R^2$$

Notație:

→ $M^K \Rightarrow \underbrace{M \dots M}_{K}$

→ $\bar{\nabla} \Rightarrow$ orice simbol cu excepția ∇

→ $R_\# \Rightarrow$ M. Turing care se deplasează dr. până la #

$$> R \circlearrowleft \bar{\#}$$

→ $L_\# \Rightarrow$ M. Turing care se deplasează stg. până la #

$$> L \circlearrowleft \bar{\#}$$

→ $R_{\bar{\#}} \Rightarrow$ M. Turing care se deplas. la dr. până la $\bar{\#}$

$$> R \circlearrowleft \#$$

→ $L_{\bar{\#}} \Rightarrow$ M. Turing care se deplas. la stg. până la $\bar{\#}$

$$> R \circlearrowleft \#$$

Ex: ? M. Turing de copiere

$$(s, \#w\underline{\#}) \vdash_M^{*} (h, \#w\#w\underline{\#})$$

$$\begin{tikzcd} > L_\# \arrow[r] & R \arrow[r, "\sigma \neq \#"] \arrow[d, "\#"] & \# R^2_\# \sigma L^2_\# \sigma \arrow[l, bend right] \\ & R_\# & \end{tikzcd}$$

$$\#abc\underline{\#} \xrightarrow{L_\#} \underline{\#}abc\# \xrightarrow{R} \#\underline{a}bc\# \xrightarrow[\#]{a\neq\#} \#\underline{\#}bc\# \xrightarrow{R^2_\#} \#\#bc\#\underline{\#} \xrightarrow{a}$$

$$\xrightarrow{a} \#\#bc\#\underline{a} \xrightarrow{L^2_\#} \#\underline{a}bc\#a \xrightarrow{R} \#a\underline{b}c\#a \xrightarrow{\sigma=b\neq\#}$$

$$\xrightarrow{\#} \#a\underline{\#}c\#a \xrightarrow{R^2_\#} \#a\#c\#a\underline{\#} \xrightarrow{b} \#a\#c\#a\underline{b} \xrightarrow{L^2_\#}$$

$$\longrightarrow \#a\underline{\#}c\#ab \xrightarrow{b} \#a\underline{b}c\#ab \xrightarrow{R} \#ab\underline{c}\#ab \xrightarrow[\#]{\sigma=c\neq\#}$$

$$\longrightarrow \#ab\#\#ab \xrightarrow{R^2_\#} \#ab\#\#ab\underline{\#} \xrightarrow{c} \#ab\#\#abc \xrightarrow{L^2_\#} \#ab\underline{\#}\#abc$$

$$\xrightarrow{c} \#ab\underline{c}\#abc \xrightarrow{R} \#abc\underline{\#}abc \xrightarrow{R_\#} \#abc\#abc\underline{\#}$$

$$\#u\#v\#w\underline{\#} \xrightarrow{C} \#u\#v\#w\#w\underline{\#}$$