Еднолентови детерминистични машини на Тюринг

Иво Стратев

12 юни 2020 г.

Дефиниция за еднолентова машина на Тюринг

$\langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$ наричаме **еднолентова машина на Тюринг**, ако

- Q е крайно множество. Множеството Q е множеството на контролните състояния на машината.
- Σ е крайна азбука, такава че $\Sigma \cap \{\rhd, \sqcup, \leftarrow, \rightarrow\} = \emptyset$. Азбуката Σ е входната азбука на машината.
- Γ е крайна множество, такова че $\Sigma \cap \Gamma = \emptyset$ и $\Gamma \cap \{\rhd, \sqcup, \leftarrow, \rightarrow\} = \emptyset$. Множеството Γ е множеството на спомагателните символи за машината.
- $s \in Q$. s е началното състояние.
- H е крайно множество, такова че $Q \cap H = \emptyset$. Множеството H е множеството на стоп състоянията.
- $\delta : Q \times (\Sigma \cup \Gamma \cup \{\rhd, \sqcup\}) \rightarrow H \cup (Q \times (\Sigma \cup \Gamma \cup \{\leftarrow, \rightarrow\}))$. δ е контролната функция. която задава програмата на машината и δ има свойството

$$(\forall p, q \in Q)(\forall x \in \Sigma \cup \Gamma \cup \{\leftarrow, \rightarrow\})(\delta(p, \rhd) = \langle q, x\rangle \implies x = \rightarrow)$$

Програма на еднолентова машина на Тюринг

Нека $\langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$ е еднолентова машина на Тюринг.

Оператори за четене и промяна на буква

Въвеждаме помощни оператори за четене от дума и промяна на буква

$$\begin{aligned}(.)[.] : \{\rhd\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^* \times \mathbb{N}_+ \rightarrow \Sigma \cup \Gamma \cup \{\sqcup, \rhd\} \\ (.)[.,\ .] : \{\rhd\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^* \times (\mathbb{N} \setminus \{0, 1\}) \times (\Sigma \cup \Gamma) \rightarrow \{\rhd\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*\end{aligned}$$

$$\alpha[i] = \begin{cases} \alpha_i & , \ i \leq |\alpha| \\ \sqcup & , \ i > |\alpha| \end{cases}$$

$$\begin{aligned} (\triangleright\alpha)[i, x] = \triangleright.(\alpha \ ! \ \langle i-1, x\rangle) \quad & where \\ \varepsilon \ ! \ \langle k+1, z\rangle &= \sqcup^k z \\ (y.\sigma) \ ! \ \langle 1, z\rangle &= z.\sigma \\ (y.\sigma) \ ! \ \langle k+2, z\rangle &= y.(\sigma \ ! \ \langle k+1, z\rangle) \end{aligned}$$

Нека пресметнем два примера

$$\begin{aligned} &\triangleright ab[6, c] = \\ &\triangleright.(ab \ ! \ \langle 5, c\rangle) = \\ &\triangleright.(a.(b \ ! \ \langle 4, c\rangle)) = \\ &\triangleright.(a.(b.(\varepsilon \ ! \ \langle 3, c\rangle))) = \\ &\triangleright.(a.(b.(\sqcup^2 c))) = \\ &\triangleright ab \sqcup \sqcup c \end{aligned}$$

$$\begin{aligned} &\triangleright abcc[3, d] = \\ &\triangleright.(abcc \ ! \ \langle 2, d\rangle) = \\ &\triangleright.(a.(bcc \ ! \ \langle 1, d\rangle)) = \\ &\triangleright.(a.(d.cc)) = \\ &\triangleright adcc \end{aligned}$$

Стъпкови програми

За всяко естествено число n въвеждаме функция δ^n
от $Q \times (\{\triangleright\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*) \times \mathbb{N}_+$ в $(Q \cup H) \times (\{\triangleright\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*)$,
която описва поведението на машината при направата на неповече от n стъпки чрез рекурсия така

$$\delta^0(q, \alpha, i) = \langle q, \alpha\rangle$$

Ако $n \in \mathbb{N}$, $q \in Q$, $\alpha \in \{\triangleright\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*$, $i \in \mathbb{N}_+$, то

- ако $\delta(q, \alpha[i]) \in H$, то $\delta^{n+1}(q, \alpha, i) = \langle \delta(q, \alpha[i]), \alpha\rangle$ (ако попаднем в стоп състояние прекратяваме изчислението);

- ако $\langle t, x\rangle = \delta(q, \alpha[i])$ и $x \in \Sigma \cup \Gamma$, то $\delta^{n+1}(q, \alpha, i) = \delta^{n}(t, \alpha[i, x], i)$ (ако получим команда за промяна на буквата на поция i извършваме тази команда, като ако се налага разшираяваме думата добавяйки необходимият брой шпации между думата и символа x);
- ако $\langle t, \rightarrow\rangle = \delta(q, \alpha[i])$, то $\delta^{n+1}(q, \alpha, i) = \delta^{n}(t, \alpha, i+1)$ (ако получим команда за отместване на курсора една позиция на дясно, то само местим курсора на дясно);
- ако $\langle t, \leftarrow\rangle = \delta(q, \alpha[i])$, то $\delta^{n+1}(q, \alpha, i) = \delta^{n}(t, \alpha, i-1)$ (ако получим команда за отместване на курсора една позиция на ляво, то само местим курсора на ляво, понеже α започва с $\triangleright$, то $i > 1$ и значи $i-1 \in \mathbb{N}_{+}$).

Нека означим езика $\{\triangleright\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^{*}$ с Λ. В сила е следното твърдение

$$(\forall q \in Q)(\forall \alpha \in \Lambda)(\forall i \in \mathbb{N}_{+})(\exists n \in \mathbb{N})(\exists h \in H)(\exists \beta \in \Lambda)$$
$$(\delta^{n}(q, \alpha, i) = \langle h, \beta\rangle \implies (\forall k \in \mathbb{N})(k \geq n \implies \delta^{k}(q, \alpha, i) = \langle h, \beta\rangle))$$

Защото е в сила следното твърдение

$$(\forall q, t \in Q)(\forall \alpha, \beta \in \Lambda)(\forall i, j \in \mathbb{N}_{+})(\forall n \in \mathbb{N})(\forall k \in \{0, 1, \ldots, n\})$$
$$(\delta^{n}(q, \alpha, i) = \delta^{k}(t, \beta, j) \implies (\forall s \in \mathbb{N})(\delta^{n+s}(q, \alpha, i) = \delta^{k+s}(t, \beta, j)))$$

Също така в сила е

$$(\forall q \in Q)(\forall \alpha, \beta \in \Lambda)(\forall i \in \mathbb{N}_{+})(\forall n \in \mathbb{N})(\forall h \in H)$$
$$(\delta^{n}(q, \alpha, i) = \langle h, \beta\rangle \implies (\exists k \in \{1, \ldots, n\})(\delta^{k}(q, \alpha, i) = \langle h, \beta\rangle))$$

Да разгледаме два примера.

Пример 1. Нека разгледаме следната еднолентова машина на Тюринг: $\langle \{s, m, e\},\ \{0, 1\},\ \emptyset,\ \delta,\ s,\ \{h, f\}\rangle$, където

$$
\begin{aligned}
\delta : \{s, m, e\} \times \{0, 1, \triangleright, \sqcup\} \to \{h, f\} \cup (\{s, m, e\} \times \{0, 1, \to, \leftarrow\}) \\
\delta(s, 0) = f \\
\delta(s, 1) = f \\
\delta(s, \triangleright) = f \\
\delta(s, \sqcup) = \langle m, 1\rangle \\
\delta(m, 0) = \langle m, \to\rangle \\
\delta(m, 1) = \langle m, \to\rangle \\
\delta(m, \triangleright) = f \\
\delta(m, \sqcup) = \langle e, 1\rangle \\
\delta(e, 0) = f \\
\delta(e, 1) = h \\
\delta(e, \triangleright) = f \\
\delta(e, \sqcup) = f
\end{aligned}
$$

Тогава

$$
\begin{aligned}
\delta^2(s, \triangleright\underline{\sqcup}10, 2) = &\quad (\text{by } \delta(s, \sqcup) = \langle m, 1\rangle) \\
\delta^1(m, \triangleright\underline{1}10, 2) = &\quad (\text{by } \delta(m, 1) = \langle m, \to\rangle) \\
\delta^0(m, \triangleright1\underline{1}0, 3) = &\quad (\text{by def of } \delta^0) \\
&\quad \langle m, \ \triangleright110\rangle
\end{aligned}
$$

Също така

$$
\begin{aligned}
\delta^9(s, \triangleright\underline{\sqcup}10, 2) = &\quad (\text{by } \delta(s, \sqcup) = \langle m, 1\rangle) \\
\delta^8(m, \triangleright\underline{1}10, 2) = &\quad (\text{by } \delta(m, 1) = \langle m, \to\rangle) \\
\delta^7(m, \triangleright1\underline{1}0, 3) = &\quad (\text{by } \delta(m, 1) = \langle m, \to\rangle) \\
\delta^6(m, \triangleright11\underline{0}, 4) = &\quad (\text{by } \delta(m, 0) = \langle m, \to\rangle) \\
\delta^5(m, \triangleright110\_, 5) = &\quad (\text{by } \delta(m, \sqcup) = \langle e, 1\rangle) \\
\delta^4(e, \triangleright110\underline{1}, 5) = &\quad (\text{by } \delta(e, 1) = h) \\
&\quad \langle h, \triangleright1101\rangle
\end{aligned}
$$

Очевидно $\delta^6(s, \triangleright \sqcup 10, 2) = \langle h, \ \triangleright1101\rangle$.

Пример 2. Нека разгледаме следната еднолентова машина на Тюринг:

$\langle \{s, m\},\ \{0, 1\},\ \emptyset,\ \delta,\ s,\ \{h, f\}\rangle$, където

$$\begin{aligned}
\delta : \{s, m\} \times \{0, 1, \rhd, \sqcup\} \to \{h, f\} \cup (\{s, m\} \times \{0, 1, \to, \leftarrow\}) \\
\delta(s, 0) = \langle m, 1\rangle \\
\delta(s, 1) = \langle m, 0\rangle \\
\delta(s, \rhd) = f \\
\delta(s, \sqcup) = h \\
\delta(m, 0) = \langle s, \to\rangle \\
\delta(m, 1) = \langle s, \to\rangle \\
\delta(m, \rhd) = f \\
\delta(m, \sqcup) = f
\end{aligned}$$

Тогава

$$\delta^1(s, (\rhd\varepsilon)_, 2) = \langle h, \rhd\varepsilon\rangle \quad (\text{by } \delta(s, \sqcup) = h\rangle)$$

$$\begin{aligned}
\delta^7(s, \rhd\underline{1}01, 2) = &\quad (\text{by } \delta(s, 1) = \langle m, 0\rangle\rangle) \\
\delta^6(m, \rhd\underline{0}01, 2) = &\quad (\text{by } \delta(m, 0) = \langle s, \to\rangle\rangle) \\
\delta^5(s, \rhd0\underline{0}1, 3) = &\quad (\text{by } \delta(s, 0) = \langle m, 1\rangle\rangle) \\
\delta^4(m, \rhd0\underline{1}1, 3) = &\quad (\text{by } \delta(m, 1) = \langle s, \to\rangle\rangle) \\
\delta^3(s, \rhd01\underline{1}, 4) = &\quad (\text{by } \delta(s, 1) = \langle m, 0\rangle\rangle) \\
\delta^2(m, \rhd01\underline{0}, 4) = &\quad (\text{by } \delta(m, 0) = \langle s, \to\rangle\rangle) \\
\delta^1(s, \rhd010_, 5) = &\quad (\text{by } \delta(m, \sqcup) = h\rangle) \\
&\quad \langle h, \rhd010\rangle
\end{aligned}$$

Така получихме $\delta^7(s, \rhd101, 2) = \langle h,\ \rhd010\rangle$.

Програма на еднолентова машина на Тюринг

Нека $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$ е еднолентова машина на Тюринг.

Нека $k \in \mathbb{N}$. Дефинираме частична функция $\Pi_k^{\mathcal{M}}$ от Σ^* в $H \times (\{\rhd\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*)$ като

$$\{\langle \alpha, r\rangle \mid \alpha \in \Sigma^* \ \&\ r \in H \times (\{\rhd\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*) \ \&\ (\exists n \in \mathbb{N}_+)(r = \delta^n(s,\ \rhd \sqcup^k \alpha,\ 2))\}$$

Няма да доказваме, че така дефинирана $\Pi_k^{\mathcal{M}}$ в действителност е функция макар и частична. Реално $\Pi_k^{\mathcal{M}}$ по дума $\alpha \in \Sigma^*$ ако има $n \in \mathbb{N}$, такова, че n-тата стъпкова програма по машината $\mathcal{M}$ завършва (изпада в стоп състояние) при стартиране на изчисление от началното състояние, при инциализирана лента, на която между възпиращият символ $\rhd$ и думата α има предварително заделена памет от k празни клетки (blank символи) и курсор сетнат на позиция 2 "връща" изчислението направено от δ^n.

Например ако разглеждаме машината от **Пример 1**, то Π_1 със сигурност е дефинирана в 10 и при това $\Pi_1(10) = \langle h, \triangleright 1101\rangle$. Дори още по-силно може да се докаже, че в този пример Π_1 е тотална функция от $\{0,1\}^*$ в $\{h\}\times(\{\triangleright 1\}\cdot\{0,1\}^*\cdot\{1\})$ действаща по правилото $\Pi_1(\alpha) = \langle h,\ \triangleright.1.\alpha.1\rangle$.

Ако разгледаме машината от **Пример 2**, то може да се докаже, че Π_0 е тотална функция с домейн $\{0,1\}^*$ действаща по правилото $\Pi_0(\alpha) = \langle h, \triangleright.\overline{\alpha}\rangle$, където ако $\alpha \in \{0,1\}^*$, то $\overline{\alpha}$ е означение за думата, която се получава от α при побуквено инвертиране. Например от направеното пресмятане $\Pi_0(101) = \langle h,\ \triangleright 010\rangle$.

Частичната функция, която изчислява дадена ЕМТ спрямо фиксирано стоп състояние

Нека $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$ е еднолентова машина на Тюринг. Нека $h \in H$. Нека $k \in \mathbb{N}$. Частичната функция, която M изчислява с начална буферна памет от k клетки спрямо стоп състояние h ще бележим с $[\mathcal{M}, h, k]$ и тя е

$$\{\langle \alpha, \beta\rangle \in \Sigma^* \times (\Sigma \cup \Gamma)^* \mid \langle \alpha, \langle h, \triangleright\beta\rangle\rangle \in \Pi_k^{\mathcal{M}}\}$$

Например ако разглеждаме машината от **Пример 1**, то

$$[\mathcal{M}, h, 1] = \{\langle \alpha, 1\alpha 1\rangle \mid \alpha \in \{0,1\}^*\}$$

Тоест $[\mathcal{M}, h, 1]$ е тотална функция и действа по правилото

$$[\mathcal{M}, h, 1](\alpha) = 1\alpha 1$$

Ако разглеждаме машината от **Пример 2**, то

$$[\mathcal{M}, h, 0] = \{\langle \alpha, \overline{\alpha}\rangle \mid \alpha \in \{0,1\}^*\}$$

Тоест $[\mathcal{M}, h, 0]$ е тотална функция и действа по правилото

$$[\mathcal{M}, h, 0](\alpha) = \overline{\alpha}$$

Дефиниция за изчислима функция между азбуки

Нека I и O са две крайни азбуки. Нека $f : I^* \to O^*$. Казваме, че f е изчислима функция между азбуки, ако съществува ЕМТ $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$, съществува $h \in H$ и съществува $k \in \mathbb{N}$ такива, че $f \subseteq [\mathcal{M}, h, k]$.

Ако $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$, $h \in H$ и $k \in \mathbb{N}$ са такива, че $f \subseteq [\mathcal{M}, h, k]$, то ясно е, че $I \subseteq \Sigma$ и $O \subseteq \Sigma \cup \Gamma$.

Дефиниция за изчислима функция между езици

Нека L е език над крайна азбука I и K е език над крайна азбука O. Нека $f : L \to K$. Казваме, че f е изчислима функция между езици, ако съществува ЕМТ $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$, съществува $h \in H$ и съществува $k \in \mathbb{N}$ такива, че $f \subseteq [\mathcal{M}, h, k]$.

Дефиниция за изчислима s-местна частична функция между естествени числа

Нека от сега до края с $\mathtt{Bin}$ означим множеството на всички правилни двоични записи на естествени числа. Тоест $\mathtt{Bin} = \{0\} \cup \{1\} \cdot \{0, 1\}^*$. Нека с B означим биекцията между $\mathbb{N}$ и $\mathtt{Bin}$, която на всяко естествено число съпоставя думата, която е неговият двоичен запис. Например $B(5) = 101$.

Нека $s \in \mathbb{N}$ и f е частична фунцкия между $\mathbb{N}^s$ и $\mathbb{N}$. Ще казваме, че f е изчислима s-местна частична функция между естествени числа, ако

$$\{\langle\ B(x_1)\$B(x_2)\$\ldots\$B(x_s)\ ,\quad B(f(x_1, x_2, \ldots, x_s))\ \rangle \mid \langle x_1, x_2, \ldots, x_s\rangle \in \mathbb{N}^s\}$$

е изчислима функция между езиците $\mathtt{Bin} \cdot (\{\$\} \cdot \mathtt{Bin})^{s-1}$ и $\mathtt{Bin}$.

Например тоталната функция $g : \mathbb{N} \to \mathbb{N}$ действаща по правилото

$$g(n) = 2^{\mathtt{Log}(n)+2} + 2n + 1$$

е изчислима, защото фунцкията $h : \mathtt{Bin} \to \mathtt{Bin}$ действаща по правилото $h(\alpha) = 1.\alpha.1$ е изчислима от машината от **Пример 1**, където

$$\mathtt{Log}(n) = \begin{cases} 0 & , n = 0 \\ \lfloor \log_2(n) \rfloor & , n \geq 1 \end{cases}$$

Разрешимост

До сега се интересувахме от стоп състоянието, в което изпада една ЕМТ както и каква дума остава на лентата след конкретен вход на лентата. Интересувайки се от думата, която остава на лентата гледахме на една ЕМТ като на **генератор**.

Можем да забравим за резултата на лентата, а да се интересуваме само от стоп състоянието, в което изпада машината при конкретен вход. Така ще гледаме на една ЕМТ като на **разпознавател**, ако тя има само две стоп състояния.

Частичната функция $\mathtt{halt}$

Нека $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ H\rangle$ е ЕМТ.

Дефинираме частична функция $\mathtt{halt}_{\mathcal{M}}$ от Σ^* в H така

$$\{\langle \alpha, h\rangle \in \Sigma^* \times H \mid (\exists \beta \in \{\rhd\} \cdot (\Sigma \cup \Gamma \cup \{\sqcup\})^*)(\langle \alpha, \langle h, \beta\rangle\rangle \in \Pi_1^{\mathcal{M}})\}$$

Дефиниция за разрешимимост от ЕМТ на език

Нека Σ е крайна азбука. Нека L е език над Σ. Нека $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ \{no, yes\}\rangle$ е ЕМТ. Казваме, че $\mathcal{M}$ разрешава L, ако са изпълнени следните три условия

- $\mathtt{halt}_{\mathcal{M}}$ е тотална функция;
- $(\forall \omega \in L)(\mathtt{halt}_{\mathcal{M}}(\omega) = yes)$;
- $(\forall \omega \in \Sigma^* \setminus L)(\mathtt{halt}_{\mathcal{M}}(\omega) = no)$.

Дефиниция за разрешим език

Нека Σ е крайна азбука. Нека L е език над Σ. Казваме, че L е разрешим език, ако L се разрешава от някоя ЕМТ. Тоест L е разрешим език ТСТК съществува ЕМТ $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ \{no, yes\}\rangle$ такава, че $\mathcal{M}$ разрешава L.

Дефиниция за полуразрешимимост от ЕМТ на език

Нека Σ е крайна азбука. Нека L е език над Σ. Нека $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ \{no, yes\}\rangle$ е ЕМТ. Казваме, че $\mathcal{M}$ полуразрешава L, ако са изпълнени следните три условия

- $L \subseteq \mathrm{Dom}(\mathtt{halt}_{\mathcal{M}})$ т.е $\mathtt{halt}_{\mathcal{M}}$ е дефинирана върху всяка дума от L;
- $(\forall \omega \in L)(\mathtt{halt}_{\mathcal{M}}(\omega) = yes)$;
- $(\forall \omega \in \Sigma^* \setminus L)(\omega \in \mathrm{Dom}(\mathtt{halt}_{\mathcal{M}}) \implies \mathtt{halt}_{\mathcal{M}}(\omega) = no)$.

Дефиниция за полуразрешим език

Нека Σ е крайна азбука. Нека L е език над Σ. Казваме, че L е полуразрешим език, ако L се полуразрешава от някоя ЕМТ. Тоест L е разрешим език ТСТК съществува ЕМТ $\mathcal{M} = \langle Q,\ \Sigma,\ \Gamma,\ \delta,\ s,\ \{no, yes\}\rangle$ такава, че $\mathcal{M}$ полуразрешава L.

Важна теорема

Нека Σ е крайна азбука. Нека L е език над Σ. Очевидно ако L е разрешим, то L е полуразрешим. Обратното не е вярно. Известно е, че **СТОП**-проблемът не е разрешим, но е полуразрешим!

Теорема. L е разрешим ТСТК L и $\Sigma^* \setminus L$ са полуразрешими езици.

Пример за разрешим език

Нека Σ е крайна азбука. Нека $L = \{\omega\#\omega \mid \omega \in \Sigma^+\}$.

Ще докажем, че L е разрешим като построим ЕМТ, която го разрешава.

Нека разгледаме няколко примера за да хванем как бихме могли да разрешим L. Нека за примерите да приемем, че $\Sigma = \{a, b, c\}$.

Първо добре е да се подсигурим, че # разделя две думи с положителна дължина. За целта трябва да прочетем цялата активна лента.

$$\triangleright\underline{\sqcup}abc\#abc$$
$$\triangleright \sqcup abc\#abc\_$$

След това да се върнем назад и да запомним видяният символ и да го изтрием. След, което да вървим наляво преминавайки през # до първата срещната буква от Σ.

$$\begin{aligned} &\triangleright \sqcup abc\#ab\underline{c} \\ c:\ &\triangleright \sqcup abc\#ab\underline{\$} \\ c:\ &\triangleright \sqcup abc\#a\underline{b}\$ \\ c:\ &\triangleright \sqcup abc\#ab\underline{\$} \\ c:\ &\triangleright \sqcup abc\underline{\#}ab\$ \end{aligned}$$

Когато я стигнем трябва да я сравним със запомнената, ако съвпада да я изтрием и да върви надясно до първият \$. Защото след него или следват други \$ или активната част свършва.

$$\begin{aligned} c:\ &\triangleright \sqcup ab\underline{c}\#ab\$ \\ &\triangleright \sqcup ab\underline{\$}\#ab\$ \\ &\triangleright \sqcup ab\$\underline{\#}ab\$ \\ &\triangleright \sqcup ab\$\#ab\underline{\$} \end{aligned}$$

След като стигнем $ се връщаме назад и повтаряме процедурата.

$$\begin{aligned}
&\rhd \sqcup \mathfrak{a}\mathfrak{b}\$\#\mathfrak{a}\underline{\mathfrak{b}}\$\\
\mathfrak{b}:\ &\rhd \sqcup \mathfrak{a}\mathfrak{b}\$\#\mathfrak{a}\underline{\$}\$\\
\mathfrak{b}:\ &\rhd \sqcup \mathfrak{a}\mathfrak{b}\$\#\underline{\mathfrak{a}}\$\$\\
\mathfrak{b}:\ &\rhd \sqcup \mathfrak{a}\mathfrak{b}\$\underline{\#}\mathfrak{a}\$\$\\
\mathfrak{b}:\ &\rhd \sqcup \mathfrak{a}\mathfrak{b}\underline{\$}\#\mathfrak{a}\$\$\\
\mathfrak{b}:\ &\rhd \sqcup \mathfrak{a}\underline{\mathfrak{b}}\$\#\mathfrak{a}\$\$\\
&\rhd \sqcup \mathfrak{a}\underline{\$}\$\#\mathfrak{a}\$\$\\
&\rhd \sqcup \mathfrak{a}\$\$\underline{\#}\mathfrak{a}\$\$\\
&\rhd \sqcup \mathfrak{a}\$\$\#\mathfrak{a}\underline{\$}\$\\
&\rhd \sqcup \mathfrak{a}\$\$\#\underline{\mathfrak{a}}\$\$\\
\mathfrak{a}:\ &\rhd \sqcup \mathfrak{a}\$\$\#\underline{\$}\$\$\\
\mathfrak{a}:\ &\rhd \sqcup \mathfrak{a}\underline{\$}\$\#\$\$\$\\
\mathfrak{a}:\ &\rhd \sqcup \underline{\mathfrak{a}}\$\$\#\$\$\$\\
&\rhd \sqcup \underline{\$}\$\$\#\$\$\$
\end{aligned}$$

След изтриването отново се движим надясно до достигането на първия $.

$$\begin{aligned}
&\rhd \sqcup \$\$\$\#\underline{\$}\$\$\\
&\rhd \sqcup \$\$\$\underline{\#}\$\$\$
\end{aligned}$$

Връщайки се назад обаче вместо буква от Σ виждаме #, което значи, че трябва да започнем да се движим наляво и да достигнем до $\sqcup$ срещайки само $. В тази ситуация приемаме думата. Ако обаче видим буква от Σ трябва да отхвърлим, защото лявата дума е с дължина строго по-голяма от дясната и няма как да са равни! Така горе-долу знаем какво да правим ако лявата дума съвпада с дясната или е по-дълга или суфиксът ѝ с дължина дължината на дясната дума не съвпада! Остава да разгледаме ситуацията, когато дясната дума е строго по-дълга от лявата.

$$\begin{aligned}
&\rhd \sqcup \mathfrak{a}\mathfrak{b}\mathfrak{c}\#\mathfrak{a}\mathfrak{a}\mathfrak{b}\mathfrak{c}\\
&\rhd \sqcup \mathfrak{a}\mathfrak{b}\mathfrak{c}\#\mathfrak{a}\mathfrak{a}\mathfrak{b}\underline{\mathfrak{c}}\\
&\qquad\qquad\vdots\\
&\rhd \sqcup \$\$\$\#\mathfrak{a}\underline{\$}\$\$\\
&\rhd \sqcup \$\$\$\#\underline{\mathfrak{a}}\$\$\$\\
\mathfrak{a}:\ &\rhd \sqcup \$\$\$\#\underline{\$}\$\$\$\\
\mathfrak{a}:\ &\rhd \sqcup \underline{\$}\$\$\#\$\$\$\$\\
\mathfrak{a}:\ &\rhd \underline{\sqcup}\$\$\$\#\$\$\$\$
\end{aligned}$$

Оказва се, че в тази ситуация трябва да сме във състояние, в което сме запомнили конкретна буква от Σ, но след последният \$ от ляво не виждаме запомнената буква, а виждаме $\sqcup$ в тази ситауция отхвърляме!

Стъпка по стъпка ще направим ЕМТ $\langle Q,\ \Sigma \cup \{\#\},\ \Gamma,\ \delta,\ \mathtt{start},\ \{\mathtt{no}, \mathtt{yes}\}\rangle$, която да разреши L в обща ситуация, в която реално не знаем Σ. Първо от нашите разсъждения е ясно, че ще ни трябва само \$ за помощен символ, следователно $\Gamma = \{\$\}$. Постепенно ще правим преходите, така че директно ще получим δ. Реално остава само Q.

Ще даваме само положитлните преходи, всички други, които са необходими за да стане δ тотална функция ще бъдат $\mathtt{no}$. Започваме от преходите, с които да прочетем активната част на лентата и да се уверим, че тя е от вида $\rhd\sqcup\alpha\#\beta$, където $|\alpha| > 0$ и $|\beta| > 0$.

$$\begin{aligned}
\delta(\mathtt{start}, \sqcup) &= \langle \mathtt{left}, \rightarrow\rangle \\
(\forall x \in \Sigma)(\delta(\mathtt{left}, x) &= \langle \mathtt{mid}, \rightarrow\rangle) \\
\delta(\mathtt{mid}, \#) &= \langle \mathtt{right}, \rightarrow\rangle \\
(\forall x \in \Sigma)(\delta(\mathtt{mid}, x) &= \langle \mathtt{mid}, \rightarrow\rangle) \\
(\forall x \in \Sigma)(\delta(\mathtt{right}, x) &= \langle \mathtt{seek}, \rightarrow\rangle) \\
\delta(\mathtt{seek}, \sqcup) &= \langle \mathtt{read}, \leftarrow\rangle \\
(\forall x \in \Sigma)(\delta(\mathtt{seek}, x) &= \langle \mathtt{seek}, \rightarrow\rangle)
\end{aligned}$$

С горните преходи си подсигуряваме точно вида на лентата, който ни трябва на първо четене и гарантирано, че ще сме в състояние $\mathtt{read}$. В състояние $\mathtt{read}$, трябва да прочетем и запомним буква от дясно на $\#$. След, което да минем наляво пропускай всички букви отдясно, $\#$ и евентуалните \$ от ляво. Като ясно е, че по някакъв начин като минем през $\#$ трябва да знаем, че трябва да минем във фаза matching. Тоест не трябва да преминаваме през символите на Σ, а като стигнем до първия да го сравним със запомненият.

$$\begin{aligned}
(\forall x \in \Sigma)(\delta(\mathtt{read}, x) &= \langle \mathtt{read}_x, \$\rangle) \\
(\forall x \in \Sigma)(\delta(\mathtt{read}_x, \$) &= \langle \mathtt{read}_x, \leftarrow\rangle) \\
(\forall x \in \Sigma)(\forall y \in \Sigma)(\delta(\mathtt{read}_x, y) &= \langle \mathtt{read}_x, \leftarrow\rangle) \\
(\forall x \in \Sigma)(\delta(\mathtt{read}_x, \#) &= \langle \mathtt{match}_x, \leftarrow\rangle \\
(\forall x \in \Sigma)(\delta(\mathtt{match}_x, \$) &= \langle \mathtt{match}_x, \leftarrow\rangle \\
(\forall x \in \Sigma)(\delta(\mathtt{match}_x, x) &= \langle \mathtt{cross}, \$\rangle)
\end{aligned}$$

След като успешно match-нем буквите изтриваме, заменяйки с \$. След това, трябва да вървим надясно и да пресечем $\#$ където имаме два варианта: да видим буква или \$. Ако видим \$ значи сме изчерпали думата от дясно и трябва да вървим на ляво и за да приемем да видим \$ последниван/и от $\sqcup$. Ако видим буква от Σ трябва да направим поне още една итерация.

$$\begin{aligned}
\delta(\mathsf{cross}, \$) &= \langle \mathsf{cross}, \rightarrow \rangle \\
\delta(\mathsf{cross}, \#) &= \langle \mathsf{scout}, \rightarrow \rangle \\
\delta(\mathsf{scout}, \$) &= \langle \mathsf{finish}, \leftarrow \rangle \\
(\forall x \in \Sigma)(\delta(\mathsf{scout}, x) &= \langle \mathsf{seek}, \rightarrow \rangle) \\
\delta(\mathsf{seek}, \$) &= \langle \mathsf{read}, \leftarrow \rangle \\
\delta(\mathsf{finish}, \#) &= \langle \mathsf{finish}, \leftarrow \rangle \\
\delta(\mathsf{finish}, \$) &= \langle \mathsf{finish}, \leftarrow \rangle \\
\delta(\mathsf{finish}, \sqcup) &= \mathsf{yes}
\end{aligned}$$

Нека видим, че думата $\mathtt{ab\#ab}$ ще бъде приета.

$$\begin{aligned}
\delta^{33}(\mathtt{start}, \rhd\underline{\sqcup}\mathtt{ab\#ab}, 2) = &\quad (\text{by } \delta(\mathtt{start}, \sqcup) = \langle \mathtt{left}, \rightarrow\rangle)\\
\delta^{32}(\mathtt{left}, \rhd\sqcup\underline{\mathtt{a}}\mathtt{b\#ab}, 3) = &\quad (\text{by } \delta(\mathtt{left}, \mathtt{a}) = \langle \mathtt{mid}, \rightarrow\rangle)\\
\delta^{31}(\mathtt{mid}, \rhd\sqcup\mathtt{a}\underline{\mathtt{b}}\mathtt{\#ab}, 4) = &\quad (\text{by } \delta(\mathtt{mid}, \mathtt{b}) = \langle \mathtt{mid}, \rightarrow\rangle)\\
\delta^{31}(\mathtt{mid}, \rhd\sqcup\mathtt{ab}\underline{\#}\mathtt{ab}, 5) = &\quad (\text{by } \delta(\mathtt{mid}, \#) = \langle \mathtt{right}, \rightarrow\rangle)\\
\delta^{30}(\mathtt{right}, \rhd\sqcup\mathtt{ab\#}\underline{\mathtt{a}}\mathtt{b}, 6) = &\quad (\text{by } \delta(\mathtt{right}, \mathtt{a}) = \langle \mathtt{seek}, \rightarrow\rangle)\\
\delta^{29}(\mathtt{seek}, \rhd\sqcup\mathtt{ab\#a}\underline{\mathtt{b}}, 7) = &\quad (\text{by } \delta(\mathtt{seek}, \mathtt{b}) = \langle \mathtt{seek}, \rightarrow\rangle)\\
\delta^{28}(\mathtt{seek}, \rhd\sqcup\mathtt{ab\#ab}_, 8) = &\quad (\text{by } \delta(\mathtt{seek}, \sqcup) = \langle \mathtt{read}, \leftarrow\rangle)\\
\delta^{27}(\mathtt{read}, \rhd\sqcup\mathtt{ab\#a}\underline{\mathtt{b}}, 7) = &\quad (\text{by } \delta(\mathtt{read}, \mathtt{b}) = \langle \mathtt{read_b}, \$\rangle)\\
\delta^{26}(\mathtt{read_b}, \rhd\sqcup\mathtt{ab\#a}\underline{\$}, 7) = &\quad (\text{by } \delta(\mathtt{read_b}, \$) = \langle \mathtt{read_b}, \leftarrow\rangle)\\
\delta^{25}(\mathtt{read_b}, \rhd\sqcup\mathtt{ab\#}\underline{\mathtt{a}}\$, 6) = &\quad (\text{by } \delta(\mathtt{read_b}, \mathtt{a}) = \langle \mathtt{read_b}, \leftarrow\rangle)\\
\delta^{24}(\mathtt{read_b}, \rhd\sqcup\mathtt{ab}\underline{\#}\mathtt{a}\$, 5) = &\quad (\text{by } \delta(\mathtt{read_b}, \#) = \langle \mathtt{match_b}, \leftarrow\rangle)\\
\delta^{23}(\mathtt{match_b}, \rhd\sqcup\mathtt{a}\underline{\mathtt{b}}\mathtt{\#a}\$, 4) = &\quad (\text{by } \delta(\mathtt{match_b}, \mathtt{b}) = \langle \mathtt{cross}, \$\rangle)\\
\delta^{22}(\mathtt{cross}, \rhd\sqcup\mathtt{a}\underline{\$}\mathtt{\#a}\$, 4) = &\quad (\text{by } \delta(\mathtt{cross}, \$) = \langle \mathtt{cross}, \rightarrow\rangle)\\
\delta^{21}(\mathtt{cross}, \rhd\sqcup\mathtt{a}\$\underline{\#}\mathtt{a}\$, 5) = &\quad (\text{by } \delta(\mathtt{cross}, \#) = \langle \mathtt{scout}, \rightarrow\rangle)\\
\delta^{21}(\mathtt{cross}, \rhd\sqcup\mathtt{a}\$\underline{\#}\mathtt{a}\$, 5) = &\quad (\text{by } \delta(\mathtt{cross}, \#) = \langle \mathtt{scout}, \rightarrow\rangle)\\
\delta^{20}(\mathtt{scout}, \rhd\sqcup\mathtt{a}\$\#\underline{\mathtt{a}}\$, 6) = &\quad (\text{by } \delta(\mathtt{scout}, \mathtt{a}) = \langle \mathtt{seek}, \rightarrow\rangle)\\
\delta^{19}(\mathtt{seek}, \rhd\sqcup\mathtt{a}\$\#\mathtt{a}\underline{\$}, 7) = &\quad (\text{by } \delta(\mathtt{seek}, \$) = \langle \mathtt{read}, \leftarrow\rangle)\\
\delta^{18}(\mathtt{read}, \rhd\sqcup\mathtt{a}\$\#\underline{\mathtt{a}}\$, 5) = &\quad (\text{by } \delta(\mathtt{read}, \mathtt{a}) = \langle \mathtt{read_a}, \$\rangle)\\
\delta^{17}(\mathtt{read_a}, \rhd\sqcup\mathtt{a}\$\#\underline{\$}\$, 5) = &\quad (\text{by } \delta(\mathtt{read_a}, \$) = \langle \mathtt{read_a}, \leftarrow\rangle)\\
\delta^{16}(\mathtt{read_a}, \rhd\sqcup\mathtt{a}\$\underline{\#}\$\$, 4) = &\quad (\text{by } \delta(\mathtt{read_a}, \#) = \langle \mathtt{match_a}, \leftarrow\rangle)\\
\delta^{15}(\mathtt{match_a}, \rhd\sqcup\mathtt{a}\underline{\$}\#\$\$, 4) = &\quad (\text{by } \delta(\mathtt{match_a}, \$) = \langle \mathtt{match_a}, \leftarrow\rangle)\\
\delta^{14}(\mathtt{match_a}, \rhd\sqcup\underline{\mathtt{a}}\$\#\$\$, 3) = &\quad (\text{by } \delta(\mathtt{match_a}, \mathtt{a}) = \langle \mathtt{cross}, \$\rangle)\\
\delta^{13}(\mathtt{cross}, \rhd\sqcup\underline{\$}\$\#\$\$, 3) = &\quad (\text{by } \delta(\mathtt{cross}, \$) = \langle \mathtt{cross}, \rightarrow\rangle)\\
\delta^{12}(\mathtt{cross}, \rhd\sqcup\$\underline{\$}\#\$\$, 4) = &\quad (\text{by } \delta(\mathtt{cross}, \$) = \langle \mathtt{cross}, \rightarrow\rangle)\\
\delta^{11}(\mathtt{cross}, \rhd\sqcup\$\$\underline{\#}\$\$, 5) = &\quad (\text{by } \delta(\mathtt{cross}, \#) = \langle \mathtt{scout}, \rightarrow\rangle)\\
\delta^{10}(\mathtt{scout}, \rhd\sqcup\$\$\#\underline{\$}\$, 6) = &\quad (\text{by } \delta(\mathtt{scout}, \$) = \langle \mathtt{finish}, \leftarrow\rangle)\\
\delta^{9}(\mathtt{finish}, \rhd\sqcup\$\$\underline{\#}\$\$, 5) = &\quad (\text{by } \delta(\mathtt{finish}, \#) = \langle \mathtt{finish}, \leftarrow\rangle)\\
\delta^{8}(\mathtt{finish}, \rhd\sqcup\$\underline{\$}\#\$\$, 4) = &\quad (\text{by } \delta(\mathtt{finish}, \$) = \langle \mathtt{finish}, \leftarrow\rangle)\\
\delta^{7}(\mathtt{finish}, \rhd\sqcup\underline{\$}\$\#\$\$, 3) = &\quad (\text{by } \delta(\mathtt{finish}, \$) = \langle \mathtt{finish}, \leftarrow\rangle)\\
\delta^{6}(\mathtt{finish}, \rhd\underline{\sqcup}\$\$\#\$\$, 2) = &\quad (\text{by } \delta(\mathtt{finish}, \sqcup) = \mathtt{yes})\\
&\quad \langle \mathtt{yes}, \rhd\underline{\sqcup}\$\$\#\$\$\rangle
\end{aligned}$$

Следователно $\mathtt{halt}(\mathtt{ab\#ab}) = \mathtt{yes}$.

Задачи за упражнение

Задача 1.

Ако $n = |\Sigma|$ намерете $|Q|$.

Задача 2.

Ако $n = |\Sigma|$ определете процентът на полезни преходи, който се пресмята по формулата

$$\frac{\text{брой преходи необходими за приемане на дума}}{\text{общ брой преходи}}$$

Пресметнете приблизителната стойност за $n = 10$.

Например процентът полезни преходи (в случая на генератор, такива необходими за достигане до h) за машината от **Пример 1** е $\frac{5}{12}$, а за машината от **Пример 2** е $\frac{5}{8}$. Тоест приблизително 42% и 62.5%.

Задача 3.

Ако $\alpha \in \Sigma$ и $n = |\alpha|$ пресметнете за колко на брой стъпки построената ЕМТ приема думата $\alpha\#\alpha$.

Задача 4.

Нека $M \in \mathcal{P}(\mathbb{N})$ казваме, че M е разрешимо множество, ако езикът $B[M] = \{B(n) \mid n \in M\} \subseteq Bin$ е разрешим. Покажете ЕМТ, която разрешава множеството $\{2n \mid n \in \mathbb{N}\}$.

Задача 5.

Покажате ЕМТ, която изчислява функцията $f : \mathbb{N} \to \mathbb{N}$, действаща по правилото $f(n) = 2n + 1$.