# Poznámky k cvičeniu č. 10

Peter Kostolányi

29. novembra 2017

## Turingove stroje

**Definícia 1.** *Deterministický Turingov stroj* je šestica $A = (K, \Sigma, \Gamma, \delta, q_0, F)$, kde $K$ je neprázdna konečná množina stavov, $\Sigma$ je vstupná abeceda, $\Gamma \supseteq \Sigma$ je pracovná abeceda obsahujúca špeciálny symbol $\mathbf{B} \in \Gamma$ („blank“), $\delta\colon K \times \Gamma \nrightarrow K \times (\Gamma - \{\mathbf{B}\}) \times \{-1, 0, 1\}$ je *čiastočná* prechodová funkcia, $q_0 \in K$ je počiatočný stav a $F \subseteq K$ je množina akceptačných stavov.

*Poznámka* 1. Čiastočnosť prechodovej funkcie $\delta$ znamená, že jej výstup môže byť pre niektoré dvojice $(q, c) \in K \times \Gamma$ aj nedefinovaný: táto skutočnosť sa niekedy zvykne označovať ako $\delta(q, c) = \perp$. Symbol $\nrightarrow$, použitý v uvedenej definícii, sa občas používa namiesto klasickej šípky na zdôraznenie čiastočnosti funkcie. Alternatívne sa deterministický Turingov stroj niekedy definuje aj ako nedeterministický Turingov stroj, kde pre každé $q \in K$ a $c \in \Gamma$ je $\delta(q, c)$ najviac jednoprvková množina. Aj keď ide o formálne odlišnú definíciu, korešpondencia s našou definíciou je asi očividná.

*Poznámka* 2. Často sa vyskytuje definícia deterministického Turingovho stroja, v ktorej $\mathbf{B} \notin \Gamma$. V takom prípade sa definuje $\delta\colon K \times (\Gamma \cup \{\mathbf{B}\}) \nrightarrow K \times \Gamma \times \{-1, 0, 1\}$.

Podobne ako v prípade konečných a zásobníkových automatov, rozumieme aj pre Turingove stroje pod konfiguráciou objekt nesúci informáciu o dosiaľ vykonanej časti výpočtu postačujúcu na to, aby bolo možné iba na základe tejto informácie vo výpočte pokračovať. Z toho je zrejmé, že v konfigurácii musí byť nejakým spôsobom zakódovaný stav, obsah pásky, ako aj pozícia čítacej hlavy. Objekt nesúci takúto informáciu možno definovať množstvom spôsobov, pričom v rôznych situáciách môžu byť výhodné rôzne formalizácie. Z tohto dôvodu konfiguráciu deterministického Turingovho stroja definujeme hneď v troch variantoch, pričom neskôr budeme vždy pracovať s variantom, ktorý sa bude javiť ako momentálne najvhodnejší.

**Definícia 2.** Nech $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ je deterministický Turingov stroj. *Konfiguráciou* stroja $A$ nazveme:

**Variant 1:** Trojicu $(q, w, n) \in K \times \{\mathbf{B}\}(\Gamma - \{\mathbf{B}\})^*\{\mathbf{B}\} \times \mathbb{N}$, kde $q$ je stav, $w$ je slovo reprezentujúce obsah zapísanej časti pásky s dvoma susednými „blank“ symbolmi a $n \in \{0, \ldots, |w| + 1\}$ je pozícia čítacej hlavy vzhľadom na slovo $w$.

**Variant 2:** Dvojicu $(q, w) \in K \times \big(\{\upharpoonleft \mathbf{B}\}(\Gamma - \{\mathbf{B}\})^*\{\mathbf{B}\} \cup \{\mathbf{B}\}(\Gamma - \{\mathbf{B}\})^*\{\upharpoonleft\}(\Gamma - \{\mathbf{B}\})^*\{\mathbf{B}\}\big)$, kde $q$ je stav a $w$ je slovo reprezentujúce obsah zapísanej časti pásky s dvoma susednými „blank“ symbolmi a pozíciou hlavy vyznačenou špeciálnym symbolom $\upharpoonleft$ (hlava je na políčku nasledujúcom za týmto symbolom).

**Variant 3:** Slovo $w \in K\{\mathbf{B}\}(\Gamma - \{\mathbf{B}\})^*\{\mathbf{B}\} \cup \{\mathbf{B}\}(\Gamma - \{\mathbf{B}\})^* K (\Gamma - \{\mathbf{B}\})^*\{\mathbf{B}\}$ (za predpokladu, že $K \cap \Gamma = \emptyset$). Špeciálny symbol $\upharpoonleft$ z variantu 2 je tu nahradený priamo stavom, ktorý potom nie je nutné udržiavať ako samostatnú zložku.

*Príklad* 1. Uvažujme napríklad konfiguráciu nejakého Turingovho stroja, ktorý je v stave $q$, na páske má zapísané slovo $aaba$ a ktorého hlava číta druhý symbol $a$. Vo variante 1 by sme túto konfiguráciu zapísali ako trojicu $(q, \mathbf{B}aaba\mathbf{B}, 2)$, vo variante 2 by išlo o dvojicu $(q, \mathbf{B}a{\upharpoonleft}aba\mathbf{B})$ a vo variante 3 zodpovedá tejto konfigurácii slovo $\mathbf{B}aqaba\mathbf{B}$. V prípade, že je hlava na pozícii prvého symbolu „blank“ naľavo od zapísanej časti pásky, konfiguráciu by sme vo variante 1 zapísali ako $(q, \mathbf{B}aaba\mathbf{B}, 0)$, vo variante 2 ako $(q, {\upharpoonleft}\mathbf{B}aaba\mathbf{B})$ a vo variante 3 ako $q\mathbf{B}aaba\mathbf{B}$.

*Poznámka* 3. Symboly $\mathbf{B}$ ohraničujúce zapísanú časť pásky sú v definícii konfigurácie skutočne nutné. Keď totiž hlava Turingovho stroja spraví z prvého zapísaného políčka pásky krok doľava (alebo z posledného zapísaného políčka krok doprava), bude v ďalšej konfigurácii hlava čítať práve niektorý z týchto symbolov $\mathbf{B}$. Nikdy sa však nemôže stať, že by hlava čítala iné nezapísané políčko: prechodová funkcia $\delta$ je totiž definovaná tak, že stroj nikdy nemôže na pásku zapísať $\mathbf{B}$, a to ani vtedy, keď bol aj pôvodný obsah políčka $\mathbf{B}$. Čítacia hlava sa tak nikdy nemôže „vzdialiť“ od zapísanej časti pásky. Turingov stroj možno ekvivalentne definovať aj tak, aby mal možnosť zapisovať $\mathbf{B}$; konfiguráciu takéhoto stroja by však bolo nutné definovať opatrnejšie.

**Definícia 3.** Nech $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ je deterministický Turingov stroj. *Krok výpočtu* stroja $A$ je binárna relácia $\vdash_A$ na konfiguráciách stroja $A$ definovaná nasledovne:

($i$) Pre všetky $p, q \in K$, $n \in \mathbb{N}$, $a_1, \ldots, a_n, c \in \Gamma - \{\mathbf{B}\}$, $k \in \{1, \ldots, n\}$ a $d \in \{-1, 0, 1\}$ platí

$$(p, \mathbf{B}a_1 \ldots a_{k-1}a_k a_{k+1} \ldots a_n\mathbf{B}, k) \vdash_A (q, \mathbf{B}a_1 \ldots a_{k-1}ca_{k+1} \ldots a_n\mathbf{B}, k + d)$$

práve vtedy, keď $\delta(p, a_k) = (q, c, d)$.

($ii$) Pre všetky $p, q \in K$, $n \in \mathbb{N}$, $a_1, \ldots, a_n, c \in \Gamma - \{\mathbf{B}\}$ a $d \in \{-1, 0, 1\}$ platí

$$(p, \mathbf{B}a_1 \ldots a_n\mathbf{B}, 0) \vdash_A (q, \mathbf{B}ca_1 \ldots a_n\mathbf{B}, d + 1)$$

práve vtedy, keď $\delta(p, \mathbf{B}) = (q, c, d)$.

($iii$) Pre všetky $p, q \in K$, $n \in \mathbb{N}$, $a_1, \ldots, a_n, c \in \Gamma - \{\mathbf{B}\}$ a $d \in \{-1, 0, 1\}$ platí

$$(p, \mathbf{B}a_1 \ldots a_n\mathbf{B}, n + 1) \vdash_A (q, \mathbf{B}a_1 \ldots a_nc\mathbf{B}, n + 1 + d)$$

práve vtedy, keď $\delta(p, \mathbf{B}) = (q, c, d)$.

($iv$) V relácii $\vdash_A$ nie sú žiadne ďalšie dvojice konfigurácií.

V prípade, že je stroj $A$ zrejmý z kontextu, píšeme namiesto $\vdash_A$ iba $\vdash$.

Jazyk akceptovaný Turingovým strojom sa v literatúre definuje množstvom rôznych (ale ekvivalentných) spôsobov. Podľa našej definície bude stroj akceptovať svoj vstup, kedykoľvek sa na ňom dokáže dostať do akceptačného stavu (bez ohľadu na to, či dočítal celý svoj vstup).

**Definícia 4.** Nech $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ je deterministický Turingov stroj. *Jazyk* akceptovaný strojom $A$ je definovaný ako

$$L(A) = \{w \in \Sigma^* \mid \exists q \in F\ \exists u \in \Gamma^*\ \exists k \in \mathbb{N} : (q_0, \mathbf{B}w\mathbf{B}, 1) \vdash^* (q, \mathbf{B}u\mathbf{B}, k)\}.$$

*Poznámka* 4. Podľa predchádzajúcej definície Turingov stroj akceptuje svoj vstup aj v prípade, že výpočet cez akceptačnú konfiguráciu iba „prejde“, pričom skončí v neakceptačnej konfigurácii alebo neskončí vôbec (pod koncom výpočtu tu máme na mysli situáciu, keď sa stroj „zasekne“, pretože na daný symbol nemá z daného stavu definovaný žiaden prechod). Ľahko však vidieť, že naša definícia akceptovaného jazyka je ekvivalentná silnejšej definícii, v ktorej by sme požadovali, aby výpočet skončil („zasekol sa“) v akceptačnom stave. Ak totiž výpočet spĺňa takúto silnejšiu podmienku akceptácie, triviálne spĺňa aj podmienku z definície 4. Pre dôkaz opačným smerom si stačí uvedomiť, že prechody vedúce z akceptačných stavov sú zbytočné: ak sa stroj raz dostane do niektorého akceptačného stavu, nemusí už vo výpočte ďalej pokračovať, pretože o akceptácii vstupu „už je rozhodnuté“. Po odobratí takýchto zbytočných prechodov teda zjavne dostávame Turingov stroj akceptujúci rovnaký jazyk podľa silnejšej definície.

*Poznámka* 5. Definíciu akceptovaného jazyka je možné ekvivalentne upraviť aj tak, aby bol stroj donútený pred akceptovaním prečítať celý vstup. Detaily konštrukcie prenechávame čitateľovi.

**Definícia 5.** *Nedeterministický Turingov stroj* je šestica $A = (K, \Sigma, \Gamma, \delta, q_0, F)$, kde $K$ je neprázdna konečná množina stavov, $\Sigma$ je vstupná abeceda, $\Gamma \supseteq \Sigma$ je pracovná abeceda obsahujúca špeciálny symbol $\mathbf{B} \in \Gamma$ („blank“), $\delta \colon K \times \Gamma \to 2^{K \times (\Gamma - \{\mathbf{B}\}) \times \{-1,0,1\}}$ je prechodová funkcia, $q_0 \in K$ je počiatočný stav a $F \subseteq K$ je množina akceptačných stavov.

Princíp nedeterminizmu je pre Turingove stroje rovnaký ako v prípade konečných automatov. Čitateľ by teda iste dokázal upraviť definície konfigurácie, kroku výpočtu a akceptovaného jazyka pre deterministické stroje tak, aby zodpovedali strojom nedeterministickým.

**Úloha 1.** Skonštruujte nedeterministický Turingov stroj pre jazyk $L = \{ww \mid w \in \{a,b\}^*\}$.

*Riešenie.* Skonštruujeme nedeterministický Turingov stroj $A = (K, \Sigma, \Gamma, \delta, q_0, F)$, ktorý bude pracovať nasledovne. Ak na začiatku výpočtu hlava číta „blank“, stroj akceptuje, pretože vstupom je prázdne slovo. V opačnom prípade si označí prvý symbol vstupného slova čiarou a zapamätá si daný symbol v stave. Následne posúva čítaciu hlavu doprava. Vždy, keď narazí na rovnaký symbol ako ten zapamätaný v stave, urobí stroj nedeterministické rozhodnutie, či už našiel začiatok druhej polovice vstupného slova (ktorá by sa v prípade akceptácie mala rovnať prvej). V takom prípade označí symbol na páske dvoma čiarami. Stroj následne nastaví čítaciu hlavu na prvý neoznačený symbol prvej časti slova a overí, či sa zhoduje s prvým neoznačeným symbolom druhej časti slova. Toto opakuje, až kým zistí nezhodu alebo chybný nedeterministický tip (v takom prípade sa zasekne) alebo úspešne overí všetky symboly prvej časti slova (v takom prípade zistí, či sú označené aj všetky symboly druhej časti a ak áno, akceptuje).

Formálne bude stroj $A$ daný nasledovne: $K = \{q_0, q_a, q_b, q_{back}, p_0, p_a, p_b, p'_a, p'_b, q_{check}, q_{acc}\}$, $\Sigma = \{a, b\}$, $\Gamma = \{a, b, a', b', a'', b'', \mathbf{B}\}$, prechodová funkcia $\delta$ bude daná ako

$$
\begin{aligned}
\delta(q_0, \mathbf{B}) &= \{(q_{acc}, a, 0)\}, & & \\
\delta(q_0, a) &= \{(q_a, a', 1)\}, & \delta(q_0, b) &= \{(q_b, b', 1)\}, \\
\delta(q_a, a) &= \{(q_a, a, 1), (q_{back}, a'', -1)\}, & \delta(q_a, b) &= \{(q_a, b, 1)\}, \\
\delta(q_b, a) &= \{(q_b, a, 1)\}, & \delta(q_b, b) &= \{(q_b, b, 1), (q_{back}, b'', -1)\}, \\
\delta(q_{back}, a'') &= \{(q_{back}, a'', -1)\}, & \delta(q_{back}, b'') &= \{(q_{back}, b'', -1)\}, \\
\delta(q_{back}, a) &= \{(q_{back}, a, -1)\}, & \delta(q_{back}, b) &= \{(q_{back}, b, -1)\}, \\
\delta(q_{back}, a') &= \{(p_0, a', 1)\}, & \delta(q_{back}, b') &= \{(p_0, b', 1)\}, \\
\delta(p_0, a) &= \{(p_a, a', 1)\}, & \delta(p_0, b) &= \{(p_b, b', 1)\}, \\
\delta(p_0, a'') &= \{(q_{check}, a'', 1)\}, & \delta(p_0, b'') &= \{(q_{check}, b'', 1)\}, \\
\delta(p_a, a) &= \{(p_a, a, 1)\}, & \delta(p_a, b) &= \{(p_a, b, 1)\}, \\
\delta(p_a, a'') &= \{(p'_a, a'', 1)\}, & \delta(p_a, b'') &= \{(p'_a, b'', 1)\}, \\
\delta(p_b, a) &= \{(p_b, a, 1)\}, & \delta(p_b, b) &= \{(p_b, b, 1)\}, \\
\delta(p_b, a'') &= \{(p'_b, a'', 1)\}, & \delta(p_b, b'') &= \{(p'_b, b'', 1)\}, \\
\delta(p'_a, a'') &= \{(p'_a, a'', 1)\}, & \delta(p'_a, b'') &= \{(p'_a, b'', 1)\}, \\
\delta(p'_a, a) &= \{(q_{back}, a'', -1)\}, & & \\
\delta(p'_b, a'') &= \{(p'_b, a'', 1)\}, & \delta(p'_b, b'') &= \{(p'_b, b'', 1)\}, \\
\delta(p'_b, b) &= \{(q_{back}, b'', -1)\}, & & \\
\delta(q_{check}, a'') &= \{(q_{check}, a'', 1)\}, & \delta(q_{check}, b'') &= \{(q_{check}, b'', 1)\}, \\
\delta(q_{check}, \mathbf{B}) &= \{(q_{acc}, a, 0)\} & &
\end{aligned}
$$

a množina akceptačných stavov bude $F = \{q_{acc}\}$. Poznamenajme ešte, že v prechodoch na prečítaný symbol $\mathbf{B}$ by sme mohli namiesto symbolu $a$ zapísať aj ľubovoľný iný symbol okrem $\mathbf{B}$ (ktorý Turingove stroje podľa ich definície zapisovať nemôžu). □

**Úloha 2.** Skonštruujte deterministický Turingov stroj pre jazyk $L = \{ww \mid w \in \{a,b\}^*\}$.

*Riešenie.* Stačí si uvedomiť, že nedeterminizmus sa v horeuvedenej konštrukcii využíva iba pri hľadaní stredu vstupného slova. To sa však dá urobiť aj deterministicky, napríklad postupným označovaním prvého a posledného neoznačeného písmena vstupu. Pokračovať potom možno rovnako ako v predchádzajúcej úlohe. Formálnu konštrukciu deterministického Turingovho stroja akceptujúceho jazyk $L$ preto prenechávame čitateľovi. □

**Ekvivalencia deterministických a nedeterministických Turingových strojov**

Podobne ako sa každý deterministický konečný automat zvykne intuitívne považovať súčasne aj za nedeterministický konečný automat, možno aj každý deterministický Turingov stroj intuitívne chápať ako špeciálny nedeterministický Turingov stroj. Táto predstava síce v ani jednom prípade nie je formálne správna, no celý problém tkvie v skutočnosti, že kým výstupom prechodovej funkcie je v deterministickom variante nejaký objekt, v nedeterministickom variante je to množina takýchto objektov.

Korektná formalizácia korešpondencie medzi deterministickým a jemu zodpovedajúcim nedeterministickým strojom je teda nasledovná. Nech $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ je deterministický Turingov stroj. Uvažujme teraz nedeterministický Turingov stroj $A' = (K, \Sigma, \Gamma, \delta', q_0, F)$, kde prechodová funkcia $\delta'$ je pre všetky $q \in K$ a $c \in \Gamma$ definovaná nasledovne: ak je hodnota $\delta(q, c)$ definovaná, tak $\delta'(q, c) = \{\delta(q, c)\}$. V opačnom prípade $\delta'(q, c) = \emptyset$. Je zrejmé, že $L(A') = L(A)$. Ku každému deterministickému Turingovmu stroju teda existuje ekvivalentný nedeterministický Turingov stroj (ktorý sa niekedy s pôvodným deterministickým strojom intuitívne stotožňuje).

Na dôkaz ekvivalencie deterministických a nedeterministických Turingových strojov teda stačí ukázať, že výpočet nedeterministického stroja možno odsimulovať aj na stroji deterministickom. To možno urobiť napríklad prehľadávaním stromu konfigurácii nedeterministického Turingovho stroja do šírky. Prehľadávanie do hĺbky použiť nemožno, keďže takýto strom konfigurácii je vo všeobecnosti nekonečný (vždy však má konečné vetvenie).

Nech teda $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ je nedeterministický Turingov stroj. Princíp deterministického Turingovho stroja $A' = (K', \Sigma', \Gamma', \delta', q_0', F')$ simulujúceho stroj $A$ možno neformálne a v hrubých rysoch opísať nasledovne:

1. Ak $A'$ dostane na vstupe slovo $w$, upraví obsah pásky na počiatočnú konfiguráciu stroja $A$ na vstupe $w$. Zapísaná časť pásky teda bude mať obsah $\overline{B}q_0w\overline{B}$, kde $\overline{B}$ je „falošný blank“. V nasledujúcom budeme obsah pásky interpretovať ako FIFO front konfigurácií stroja $A$, oddelených špeciálnym symbolom #. Symboly $\overline{B}$ pritom budeme interpretovať ako symboly **B** susediace so zapísanou časťou pásky (ktoré sú podľa definície súčasťou konfigurácie). Na začiatku simulácie je teda obsahom frontu jediná konfigurácia $\mathbf{B}q_0w\mathbf{B}$.

2. Opakuj v nekonečnom cykle:

   2.1 Ak je front prázdny, ukonči cyklus a vstup $w$ zamietni.

   2.2 V opačnom prípade, nech $C$ je prvá konfigurácia vo fronte.

   2.3 Ak je konfigurácia $C$ akceptačná (obsahuje akceptačný stav), ukonči cyklus a vstup $w$ akceptuj.

   2.4 Ak konfigurácia $C$ akceptačná nie je, pridaj na koniec frontu všetky konfigurácie $C'$ také, že $C \vdash_A C'$. Takýchto konfigurácií je vždy konečne veľa (ak je v konfigurácii $C$ stroj v stave $q$ a hlava číta písmeno $c$, ich počet je $|\delta(q, c)|$). Zmaž konfiguráciu $C$ zo začiatku frontu.

Z hľadiska implementácie na deterministickom Turingovom stroji je najmenej triviálnym krokom uvedeného algoritmu bod 2.4. Stroj môže pracovať napríklad tak, že si najprv v stave zapamätá všetky trojice z množiny $\delta(q, c)$, následne $|\delta(q, c)|$-krát skopíruje konfiguráciu $C$ na koniec frontu (pričom jednotlivé kópie pooddeľuje symbolom #) a napokon využije informácie zapamätané v stave na úpravu všetkých týchto kópií podľa trojíc z $\delta(q, c)$.