

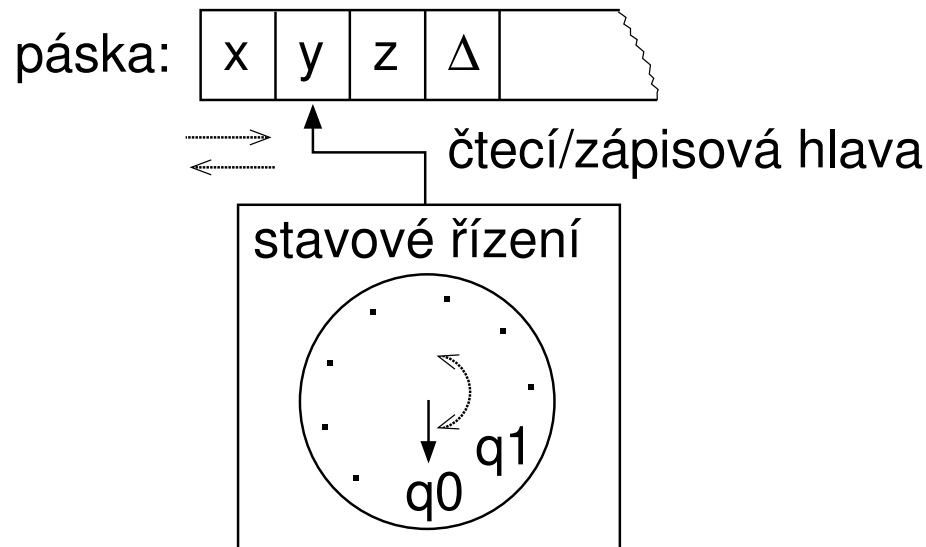
Turingovy stroje

Churchova teze

- ❖ **Churchova (Church-Turingova) teze:** *Turingovy stroje (a jim ekvivalentní systémy) definují svou výpočetní silou to, co intuitivně považujeme za efektivně vyčíslitelné.*
- ❖ Churchova teze **není teorém**, nemůžeme formálně dokazovat, že něco odpovídá našim intuitivním představám, nicméně je podpořena řadou argumentů:
 - Turingovy stroje jsou **velmi robustní** – uvidíme, že jejich různé úpravy nemění jejich výpočetní sílu (determinismus x nedeterminismus, počet pásek, ...).
 - Byla navržena řada zcela **odlišných výpočetních modelů** (λ -kalkulus, parciálně rekurzivní funkce, Minského stroje, ...), jejichž síla odpovídá Turingovým strojům.
 - Není znám **žádný výpočetní proces**, který bychom označili za efektivně vyčíslitelný a který by nebylo možné realizovat na Turingově stroji.^a

^aExistují formalizované výpočetní procesy, realizovatelné např. na TS s orákulem (nápovědou) rozhodujícím atomicky nějaký Turingovsky nerozhodnutelný problém (např. problém zastavení), které ale nepovažujeme za efektivní výpočetní procesy.

Turingův stroj



Definice 7.1 Turingův stroj (TS) je šestice tvaru $M = (Q, \Sigma, \Gamma, \delta, q_0, q_F)$, kde:

- Q je konečná množina vnitřních (řídících) stavů,
- Σ je konečná množina symbolů nazývaná vstupní abeceda, $\Delta \notin \Sigma$,
- Γ je konečná množina symbolů, $\Sigma \subset \Gamma$, $\Delta \in \Gamma$, nazývaná pásková abeceda,
- parciální funkce $\delta : (Q \setminus \{q_F\}) \times \Gamma \rightarrow Q \times (\Gamma \cup \{L, R\})$, kde $L, R \notin \Gamma$, je přechodová funkce,
- q_0 je počáteční stav, $q_0 \in Q$ a
- q_F je koncový stav, $q_F \in Q$.

Konfigurace Turingova stroje

- ❖ Symbol Δ značí tzv. **blank** (prázdný symbol), který se vyskytuje na místech pásky, která nebyla ještě použita (může ale být na pásku zapsán i později).
- ❖ **Konfigurace pásky** je dvojice sestávající z nekonečného řetězce reprezentujícího obsah pásky a pozice hlavy na tomto řetězci – přesněji jde o prvek množiny $\{\gamma\Delta^\omega \mid \gamma \in \Gamma^*\} \times \mathbb{N}$.^a
- ❖ **Konfiguraci pásky** zapisujeme jako $\Delta xyz \underline{z} \Delta x \Delta \Delta \dots$ (podtržení značí pozici hlavy).
- ❖ **Konfigurace stroje** je pak dána stavem řízení a konfigurací pásky – formálně se jedná o prvek množiny $Q \times \{\gamma\Delta^\omega \mid \gamma \in \Gamma^*\} \times \mathbb{N}$.

^a Pro libovolnou abecedu Σ je Σ^ω množina všech *nekonečných* řetězců nad Σ , tj. nekonečných posloupností symbolů ze Σ . Pro připomenutí: Σ^* je množina všech *konečných* řetězců nad Σ .

Přechodová relace TS

❖ Pro libovolný řetězec $\gamma \in \Gamma^\omega$ a číslo $n \in \mathbb{N}$ označme γ_n n -tý symbol daného řetězce a označme $s_b^n(\gamma)$ řetězec, který vznikne z γ záměnou γ_n za b .

❖ Krok výpočtu TS M definujeme jako *nejmenší* binární relaci \vdash_M takovou, že

$\forall q_1, q_2 \in Q \forall \gamma \in \Gamma^\omega \forall n \in \mathbb{N} \forall b \in \Gamma$:

- $(q_1, \gamma, n) \vdash_M (q_2, \gamma, n + 1)$ pro $\delta(q_1, \gamma_n) = (q_2, R)$ – operace **posuvu doprava** při γ_n pod hlavou,
- $(q_1, \gamma, n) \vdash_M (q_2, \gamma, n - 1)$ pro $\delta(q_1, \gamma_n) = (q_2, L)$ a $n > 0$ – operace **posuvu doleva** při γ_n pod hlavou a
- $(q_1, \gamma, n) \vdash_M (q_2, s_b^n(\gamma), n)$ pro $\delta(q_1, \gamma_n) = (q_2, b)$ – operace **zápisu** b při γ_n pod hlavou.

Výpočet TS

- ❖ Výpočet TS M začínající z konfigurace K_0 je posloupnost konfigurací K_0, K_1, K_2, \dots ,
 - ve které $K_i \vdash_M K_{i+1}$ pro všechna $i \geq 0$ taková, že K_{i+1} je v dané posloupnosti, a
 - která je buď
 - nekonečná, a nebo
 - konečná s koncovou konfigurací (q, γ, n) , přičemž rozlišujeme následující typy zastavení TS:
 1. normální – přechodem do koncového stavu, tj. $q = q_F$, a
 2. abnormální, kdy $q \neq q_F$ a:
 - (a) pro (q, γ_n) není δ definována, nebo
 - (b) hlava je na nejlevější pozici pásky a dojde k posunu doleva, tj. $n = 0$ a $\delta(q, \gamma_n) = (q', L)$ pro nějaké $q' \in Q$.

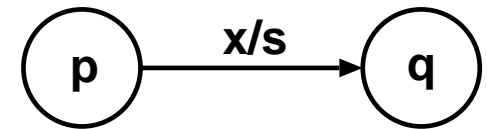
Poznámka – alternativní definice TS

❖ Používají se i některé **alternativní definice TS**, u kterých se dá snadno ukázat **vzájemná převoditelnost**:

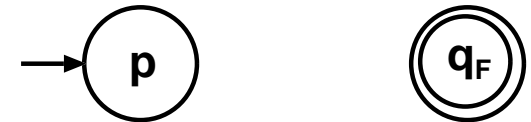
- namísto jediného q_F je povolena množina koncových stavů,
- namísto q_F je zavedena dvojice q_{accept} a q_{reject} ,
- na prvním políčku pásky je „napevno“ zapsán symbol konce pásky, z něhož není možný posun doleva,
- při zavedení obou předchozích bodů je δ obvykle definovaná jako totální funkce,
- přepis a posuv hlavy jsou spojeny do jedné operace
- apod.

Grafická reprezentace TS

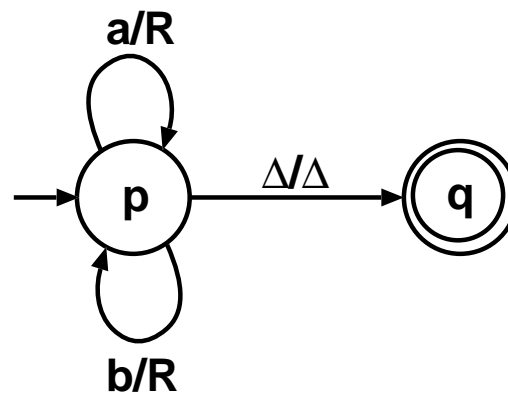
❖ Grafická reprezentace **přechodu** (x – co se čte, s – zápis/ L/R):



❖ Grafická reprezentace **počátečního** a **koncového stavu**:

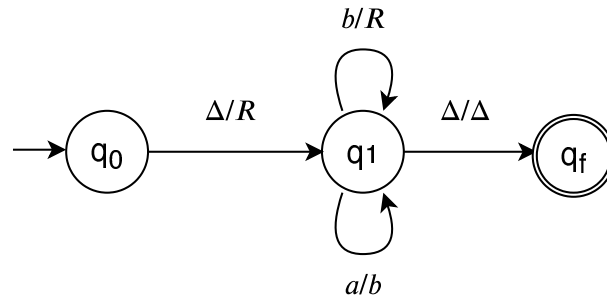


Příklad 7.1 TS, který posouvá hlavu doprava na první Δ počínaje aktuální pozicí (např. $\Delta \underline{a}ab\Delta \dots \rightarrow \Delta aab \underline{\Delta} \dots$):

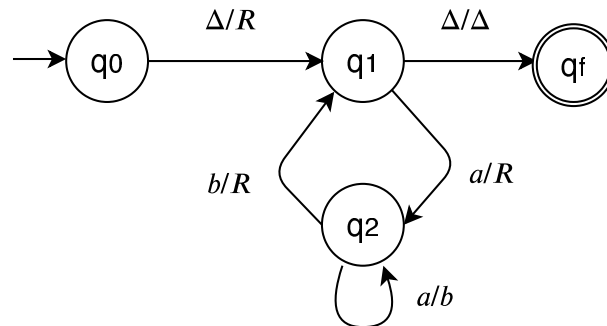


Příklady TS

Příklad 7.2 TS, který modifikuje pásku ve tvaru $\underline{\Delta}a^n\Delta$ pro $n > 0$ na $\Delta b^n \underline{\Delta}$

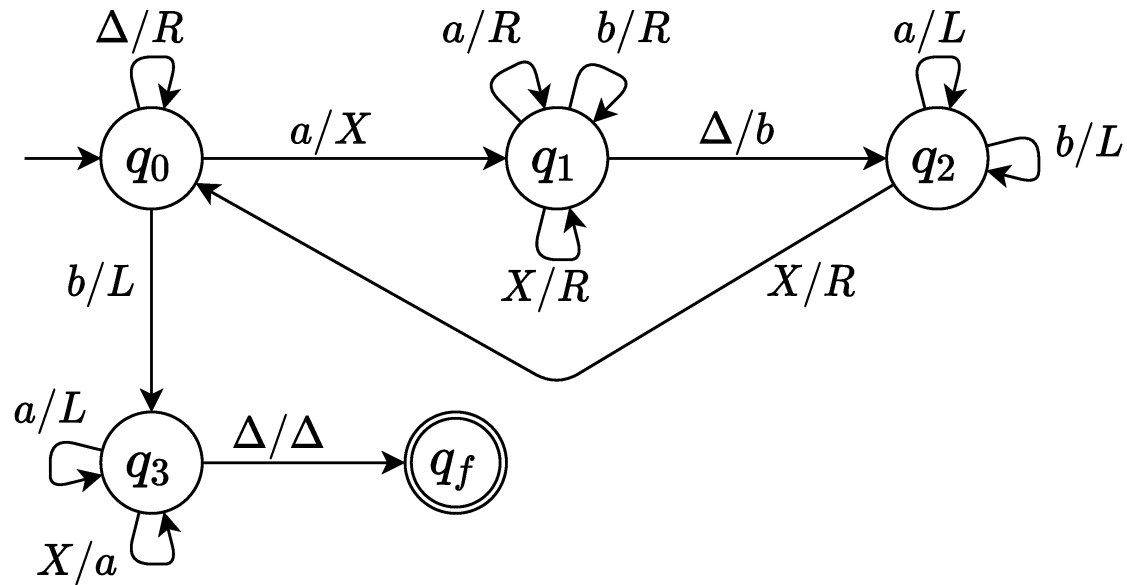


Příklad 7.3 TS, který modifikuje pásku ve tvaru $\underline{\Delta}a^{2n}\Delta$ pro $n > 0$ na $\Delta(ab)^n \underline{\Delta}$



Příklady TS

Příklad 7.4 TS, který modifikuje pásku ve tvaru $\Delta a^n \Delta$ pro $n > 0$ na $\Delta a^n b^n \Delta$



Poznámka 7.1

- Uvědomme si, že na TS lze také nahlížet jako na **výpočetní mechanismy implementující funkce $\Sigma^* \rightarrow \Gamma^*$** tím, že transformují počáteční neblankový prefix své pásky na jiný neblankový prefix při přechodu do koncového stavu.
- Vzhledem k tomu, že TS nemusí každý svůj vstup přijmout, jsou funkce jimi implementované obecně **parciální**.
- Blíže se budeme vyčíslováním funkcí TS zabývat dále.

Turingovy stroje jako akceptory jazyků

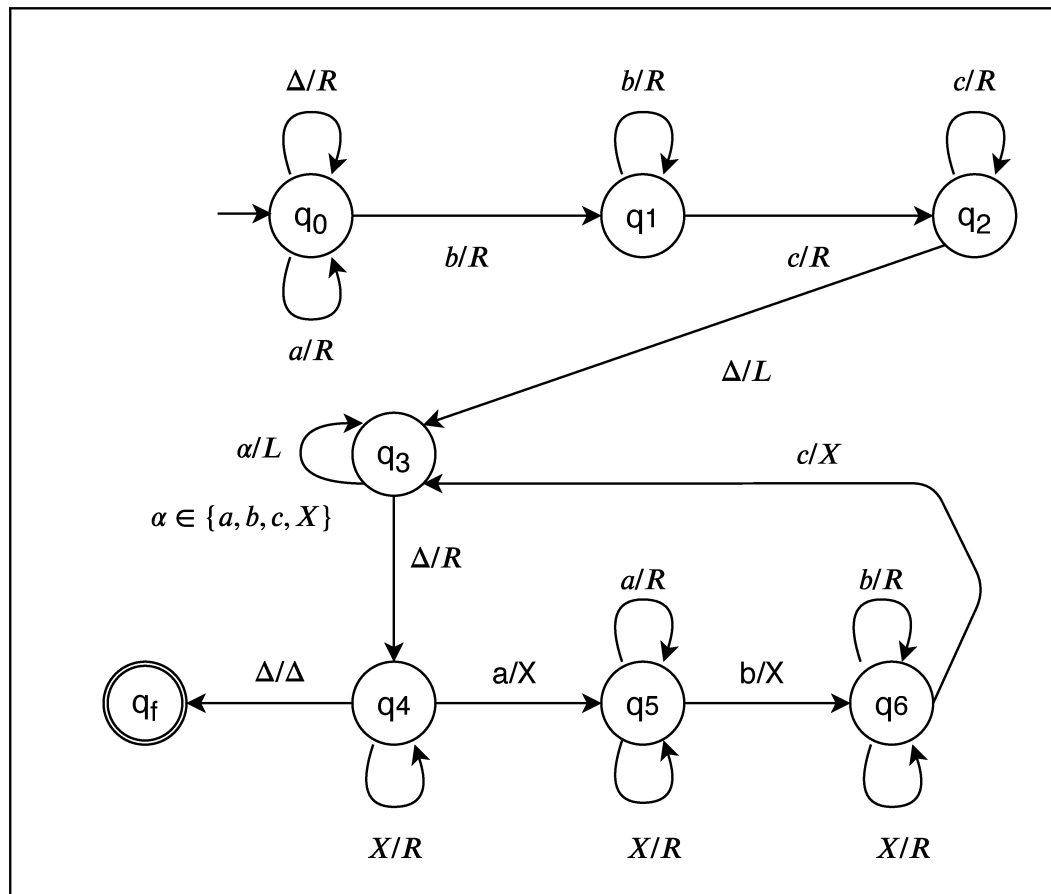
Jazyk přijímaný TS

Definice 7.2

1. Řetězec $w \in \Sigma^*$ je přijat TS $M = (Q, \Sigma, \Gamma, \delta, q_0, q_F)$, jestliže M při aktivaci z počáteční konfigurace pásky $\underline{\Delta}w\Delta\dots$ a počátečního stavu q_0 zastaví přechodem do koncového stavu q_F , tj. $(q_0, \underline{\Delta}w\Delta^\omega, 0) \stackrel{*}{\vdash}_M (q_F, \gamma, n)$ pro nějaké $\gamma \in \Gamma^*$ a $n \in \mathbb{N}$.
 2. Množinu $L(M) = \{w \mid w \text{ je přijat TS } M\} \subseteq \Sigma^*$ nazýváme **jazyk přijímaný TS M** .
- ❖ Alternativně můžeme **přijetí řetězce TS** definovat tak, že TS začíná s konfigurací pásky $\underline{\Delta}w\Delta\dots$ a zastaví s konfigurací pásky $\underline{\Delta}Y\Delta\dots$, $Y \in \Gamma \setminus \Sigma$, (Y značí Yes).

TS pro nebezkontextový jazyk

Příklad 7.5 TS, který přijímá jazyk $L = \{a^n b^n c^n \mid n > 0\}$



Vícepáskové Turingovy stroje

Vícepáskové TS

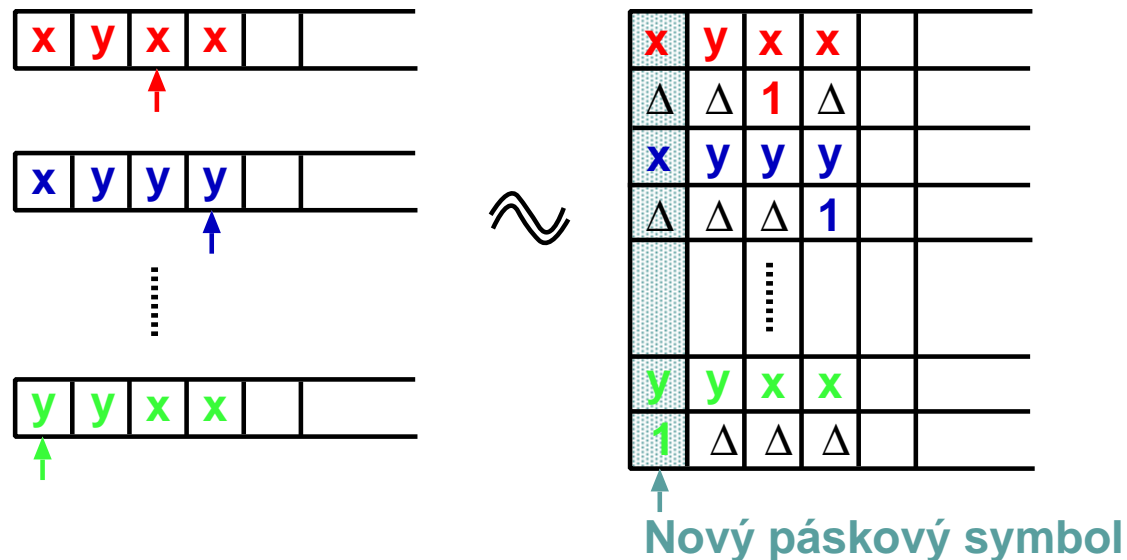
❖ Uvažujme TS, který má k pásek s páskovými $\Gamma_1, \Gamma_2, \dots, \Gamma_k$ a k odpovídajících hlav s přechodovou funkcí tvaru

$$\delta : (Q \setminus \{q_F\}) \times \Gamma_1 \times \Gamma_2 \times \dots \times \Gamma_k \longrightarrow Q \times \Gamma'_1 \times \Gamma'_2 \times \dots \times \Gamma'_k$$

kde $\Gamma'_i = \Gamma_i \cup \{L, R\}$.

Věta 7.1 Pro každý k -páskový TS M existuje jednopáskový TS M' takový, že $L(M) = L(M')$.

Důkaz. (idea)



Důkaz pokračuje dále.

Pokračování důkazu. Důkaz provedeme tak, že ukážeme algoritmus převodu na ekvivalentní jednopáskový TS:

- Předpokládáme, že **přijímaný řetězec** je u k -páskového stroje na počátku zapsán na první pásce, všechny ostatní pásy jsou prázdné a všechny hlavy jsou na nejlevější pozici.
- Původních k pásek simulujeme **rozšířením páskové abecedy o $2k$ -tice**, v nichž vždy i -tá složka pro liché i reprezentuje obsah $(\frac{i+1}{2})$ -ní pásky a na pozici $i + 1$ je Δ nebo 1 podle toho, zda se na ní nachází příslušná hlava či nikoliv.
- **Počet načítaných kombinací symbolů v původním automatu je konečný** a tudíž si výše uvedené rozšíření můžeme skutečně dovolit.
- Při simulaci k -páskového TS pak nejprve převedeme původní obsah první pásky na ekvivalentní obsah zakódovaný v $2k$ -ticích a pak každý krok simulujeme několika kroky.
- Při simulaci využíváme **stavy ve formě $(k + 1)$ -tic**, kde první složka je stav původního TS a ostatní složky jsou aktuálně čtené symboly.
- **Při rozhodování o dalším kroku** pak bereme na zřetel především tento stav, aktuální načítaný symbol není důležitý a vždy se při čtení přemístíme na speciální pozici nalevo od užitečného obsahu pásky.

Důkaz pokračuje dále.

Pokračování důkazu.

- Po rozhodnutí o dalším kroku se posunujeme postupně doprava na pozice, která se mají modifikovat, provedeme příslušnou změnu a vrátíme se zpět doleva.
- Za nový aktuální stav považujeme $(k + 1)$ -tici danou novým stavem simulovaného TS a novou k -tici reprezentující modifikace na simulovaných páskách.
- Uvědomme si, že jeden krok simulace vícepáskového stroje vyžaduje 2 průchody páskou – více viz přednáška o složitosti.
- Navíc je nutné korektně simulovat „přepadnutí“ hlavy na kterékoliv pásce a převod dosud nevyužitých míst pásky s Δ na odpovídající $2k$ -tici blank symbolů.
- Při řádné formalizaci popsaného algoritmu pak není těžké ukázat, že výsledný TS skutečně simuluje původní TS.

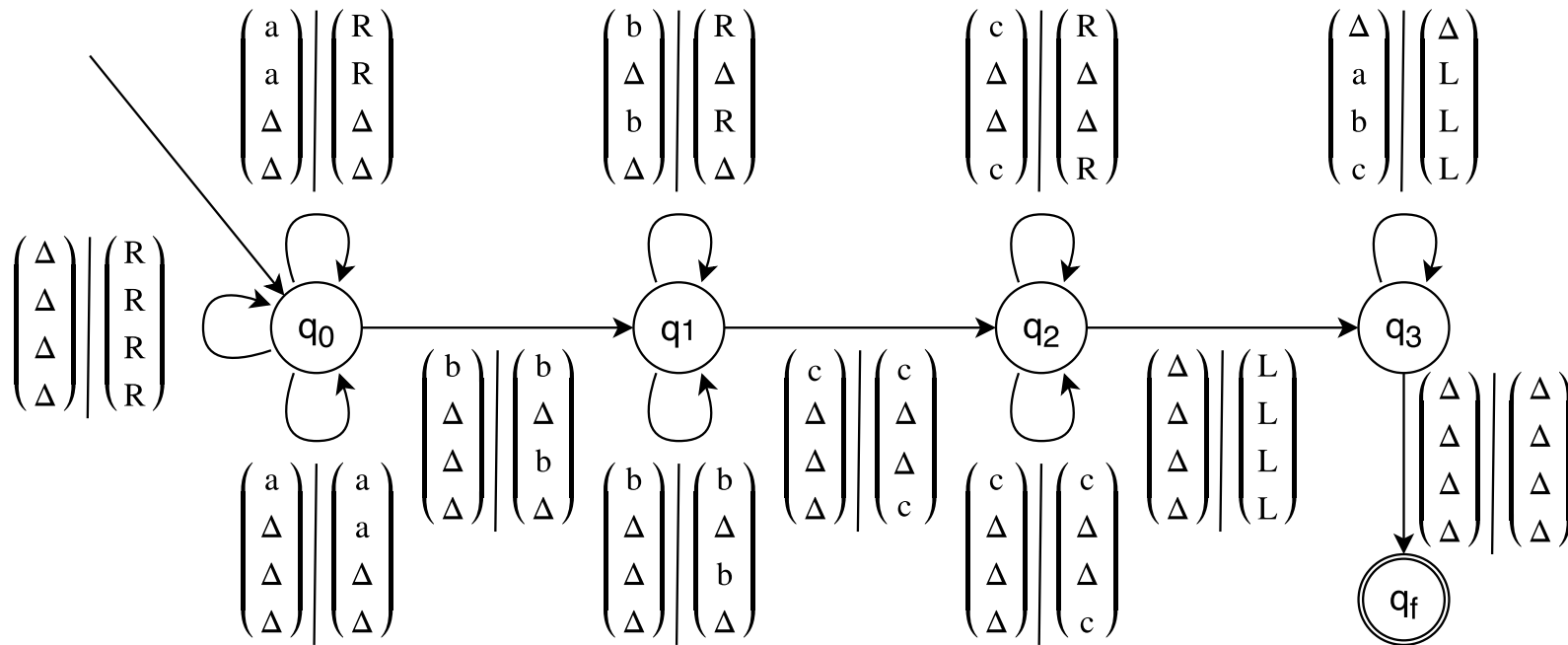
□

❖ Závěr:

Zvětšení paměťových možností TS nerozšiřuje jejich schopnosti přijímat jazyky!

Příklad vícepáskového TS

Příklad 7.6 4-páskový TS, který přijímá jazyk $L = \{a^n b^n c^n \mid n > 0\}$



Nedeterministické Turingovy stroje

Nedeterministické TS

Definice 7.3 Nedeterministický TS je šestice $M = (Q, \Sigma, \Gamma, \delta, q_0, q_F)$, kde význam jednotlivých složek je shodný s deterministickým TS až na δ , jež má tvar:

$$\delta : (Q \setminus \{q_F\}) \times \Gamma \longrightarrow 2^{Q \times (\Gamma \cup \{L, R\})}$$

Definice 7.4 Jazyk $L(M)$ NTS $M = (Q, \Sigma, \Gamma, \delta, q_0, q_F)$ je množina řetězců $w \in \Sigma^*$ takových, že M při aktivaci z q_0 při počátečním obsahu pásky $\underline{\Delta}w\Delta\dots$ **může zastavit** přechodem do q_F .

Věta 7.2 Pro každý NTS M existuje DTS M' takový, že $L(M) = L(M')$.

Důkaz. (idea)

- NTS M budeme simulovat třípáskovým DTS. Význam jednotlivých pásek tohoto stroje je následující:
 - Páska 1 obsahuje vstupní řetězec.
 - Páska 2 je pracovní páska. Obsahuje kopii pásky 1 ohraničenou vhodnými speciálními značkami. Po neúspěšném pokusu o přijetí je její obsah smazán a obnoven z první pásky.
 - Páska 3 obsahuje kódovanou volbu posloupností přechodů; při neúspěchu bude její obsah nahrazen jinou posloupností.

Důkaz pokračuje dále.

Pokračování důkazu.

- Zvolená posloupnost přechodů je kódována posloupností čísel přiřazených přechodům simulovaného stroje.
- Jednotlivé posloupnosti přechodů na pásce 3 generujeme pomocí BFS: nejprve všechny výpočty délky 1, potom všechny výpočty délky 2 atd.
- Vlastní simulace probíhá takto:
 1. Okopíruj obsah pásky 1 na pásku 2.
 2. Generuj příští posloupnost přechodů na pásce 3.
 3. Simuluj provedení posloupnosti z pásky 3 na obsahu pásky 2.
 4. Vede-li zkoumaná posloupnost do q_F simulovaného stroje, zastav – vstupní řetězec je přijat. V opačném případě smaž pásku 2 a vrať se k bodu 1.
- Není obtížné nahlédnout, že jazyk takto vytvořeného stroje odpovídá jazyku původního NTS.

□

❖ Závěr:

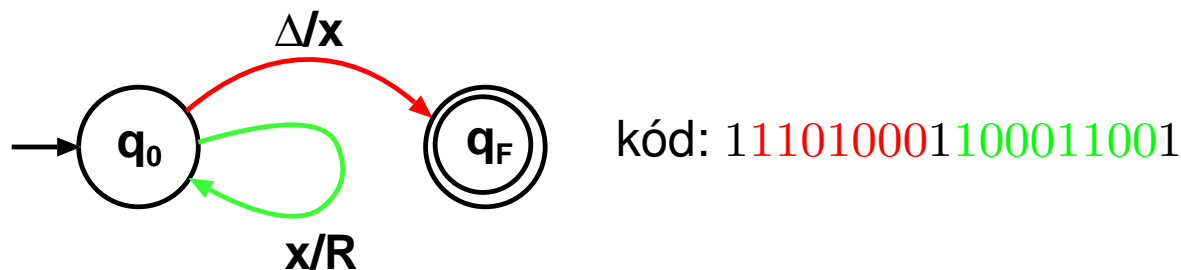
Zavedením nedeterminismu do TS se nezvyšují jejich schopnosti přijímat jazyky!

Univerzální Turingovy stroje

Kódování TS

- ❖ **Kódovací systém** pro TS zahrnuje (1) kódování stavů (tak, aby byly odlišeny všechny stavy včetně q_0 a q_F), (2) symbolů z Γ a (3) přechodové funkce δ .
- ❖ **Kódování stavů**: Množinu stavů Q uspořádáme do posloupnosti q_0, q_F, q, p, \dots, t . Stav q_j zakódujeme jako 0^j , přičemž indexujeme (např.) od nuly.
- ❖ **Kódování symbolů a příkazů L/R** : Předpokládejme, že $\Gamma = \Sigma \cup \{\Delta\}$. Uspořádáme Σ do posloupnosti a_1, a_2, \dots, a_n a zvolíme tyto kódy: $\Delta \mapsto \varepsilon$, $L \mapsto 0$, $R \mapsto 00$, $a_i \mapsto 0^{i+2}$.
- ❖ **Přechod $\delta(p, x) = (q, y)$** , kde $y \in \Gamma \cup \{L, R\}$, reprezentujeme čtveřicí (p, x, q, y) a kódujeme zřetěžením kódů p, x, q, y při použití 1 jako oddělovače, tj. jako $\langle p \rangle 1 \langle x \rangle 1 \langle q \rangle 1 \langle y \rangle$, kde $\langle _ \rangle$ značí kód $_$.
- ❖ **Celý TS** kódujeme jako posloupnost kódů přechodů oddělených a ohraničených 1.

Příklad 7.7



Univerzální TS

- ❖ Zavádí koncept „programovatelného“ stroje, který umožňuje ve vstupním řetězci specifikovat konkrétní TS (tj. program) i data, nad nimiž má tento stroj pracovat.
- ❖ TS, který má být simulován, budeme kódovat, jak bylo uvedeno na předchozí straně, vstupní řetězec budeme kódovat jako posloupnost příslušných kódů symbolů oddělených a ohraničených 1. Kód stroje a vstupního řetězce oddělíme např. #.

Příklad 7.8 TS z předchozí strany mající na vstupu xxx :

111010001100011001#1000100010001

- ❖ Univerzální TS, který zpracuje toto zadání můžeme navrhnout jako třípáskový stroj, který
 - má na 1. pásce zadání (a později výstup),
 - 2. pásku používá k simulaci pracovní pásy původního stroje a
 - na 3. pásce má zaznamenán řídicí stav simulovaného stroje a aktuální pozici hlavy (pozice hlavy i je kódována jako 0^i).

❖ Univerzální stroj pracuje takto:

1. Stroj zkontroluje, zda vstup odpovídá nějakému $M#w$ a pokud ne, abnormálně zastaví.
2. Přepíše w na 2. pásku, na 3. pásku umístí kód q_0 a za něj poznačí, že hlava se nachází na levém okraji pásky.
3. Na 2. pásce vyhledá aktuální symbol pod hlavou simulovaného stroje a na 1. pásce vyhledá přechod proveditelný ze stavu zapsaného na začátku 3. pásky pro tento vstupní symbol. Pokud žádný přechod možný není, stroj abnormálně zastaví.
4. Stroj provede na 2. a 3. pásce změny odpovídající simulovanému přechodu (přepis aktuálního symbolu, změna pozice hlavy, změna řídicího stavu).
5. Pokud nebyl dosažen stav q_F simulovaného stroje, přejdeme na bod 3. Jinak stroj vymaže 1. pásku, umístí na ní obsah 2. pásky a zastaví přechodem do svého koncového stavu.

❖ Víme, že výše uvedený stroj můžeme převést na **jednopáskový univerzální TS**, který budeme v dalším značit jako T_U .