Formální jazyky a automaty

Uzávěrové vlastnosti, rozhodnutelnost

Jan Křetínský

Fakulta informatiky, MU Brno

Jaro 2024

Rozšíření a varianty TM

- obousměrná páska

Rozšíření a varianty TM

- obousměrná páska

$\cdots$	$\sqcup$	$\sqcup$	c	a	b	a	c	c	c	a	b	c	c	$\sqcup$	$\sqcup$	$\cdots$
							$\uparrow$									

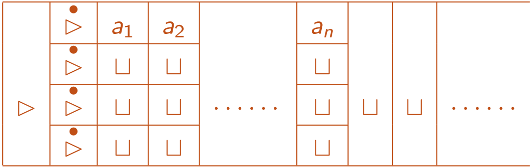
$\triangleright$	c	c	c	a	b	c	c	$\sqcup$	$\sqcup$	$\dots$
	a	b	a	c	$\sqcup$	$\sqcup$	$\sqcup$			

Rozšíření a varianty TM

- obousměrná páska
- více pásek

Rozšíření a varianty TM

- obousměrná páska
- více pásek

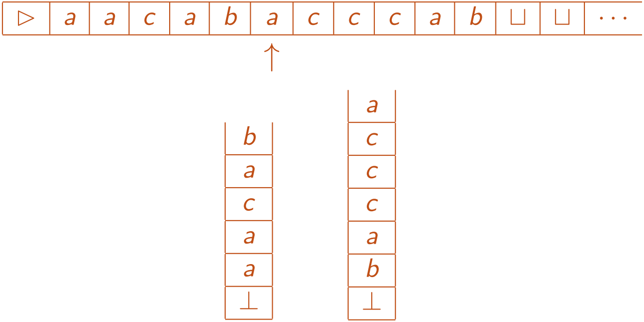


Rozšíření a varianty TM

- obousměrná páska
- více pásek
- FA se 2 zásobníky

Rozšíření a varianty TM

- obousměrná páska
- více pásek
- FA se 2 zásobníky



Rozšíření a varianty TM

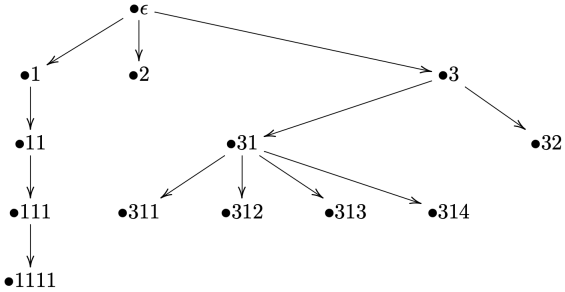
- obousměrná páska
- více pásek
- FA se 2 zásobníky
- Minského stroj (stroj se 2 počítadly)

Rozšíření a varianty TM

- obousměrná páska
- více pásek
- FA se 2 zásobníky
- Minského stroj (stroj se 2 počítadly)
- nedeterministický TM

Rozšíření a varianty TM

- obousměrná páska
- více pásek
- FA se 2 zásobníky
- Minského stroj (stroj se 2 počítadly)
- nedeterministický TM



Přehled jazykových tříd

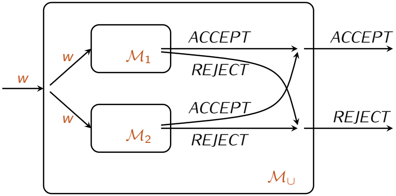
Jazyky	**Gramatiky (typ)**	**Automaty**
rekursivně spočetné (r.e.)	frázové (0)	Turingovy stroje
rekursivní	-	úplné Turingovy stroje
kontextové	kontextové (1)	lineárně ohraničené TM
bezkontextové	bezkontextové (2)	zásobníkové automaty
deterministické CFL	-	deterministické PDA
regulární	regulární (3)	konečné automaty

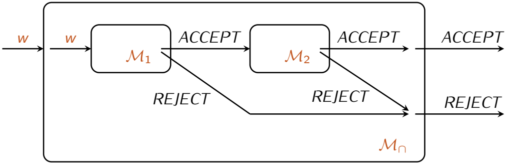
Třída na nižším řádku je vždy vlastní podtřídou třídy na vyšším řádku.

Uzávěrové vlastnosti 1/2

Věta 4.15.

Třídy rekurzivních a rekurzivně spočetných jazyků jsou uzavřeny vzhledem k operacím sjednocení, průniku, zřetězení a iteraci.





Uzávěrové vlastnosti 2/2

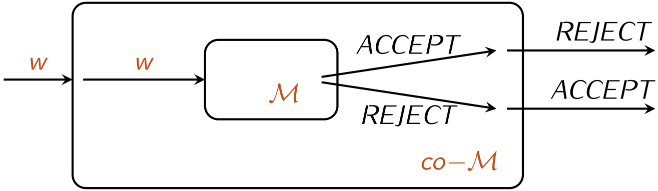
Věta 4.16.

Třída rekurzivních jazyků je uzavřena vzhledem k operaci komplementu.

Uzávěrové vlastnosti 2/2

Věta 4.16.

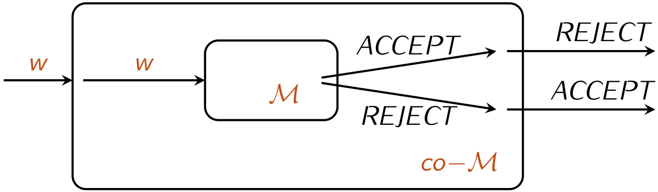
Třída rekurzivních jazyků je uzavřena vzhledem k operaci komplementu.



Uzávěrové vlastnosti 2/2

Věta 4.16.

Třída rekurzivních jazyků je uzavřena vzhledem k operaci komplementu.



Věta 4.17.

Pokud jsou L i $co{-}L$ oba rekurzivně spočetné, pak jsou oba rekurzivní.

Rozhodnutelnost problémů

Problémy jako jazyky

Problém rozhodnout, zda dané w má vlastnost P lze ztotožnit s jazykem (tj. množinou) $\{w \mid w \text{ má vlastnost } P\}$.

Příklad. Problém rozhodnout, zda daná regulární gramatika reprezentuje konečný jazyk, ztotožníme s jazykem

$$\{\mathcal{G} \mid \mathcal{G} \text{ je regulární gramatika a } L(\mathcal{G}) \text{ je konečný}\}.$$

Rozhodnutelnost problémů

Problémy jako jazyky

Problém rozhodnout, zda dané w má vlastnost P lze ztotožnit s jazykem (tj. množinou) $\{w \mid w \text{ má vlastnost } P\}$.

Příklad. Problém rozhodnout, zda daná regulární gramatika reprezentuje konečný jazyk, ztotožníme s jazykem

$$\{\mathcal{G} \mid \mathcal{G} \text{ je regulární gramatika a } L(\mathcal{G}) \text{ je konečný}\}.$$

Definice (str. 109 ve skriptech)

Problém P odpovídající jazyku $L = \{w \mid w \text{ má vlastnost } P\}$ je

- **rozhodnutelný**, právě když L je rekursivní,
- **nerozhodnutelný**, právě když L není rekursivní,
- **částečně rozhodnutelný (semirozhodnutelný)**, právě když L je rekursivně spočetný (r.e.).

Rozhodnutelnost problémů

Problémy jako jazyky

Problém rozhodnout, zda dané w má vlastnost P lze ztotožnit s jazykem (tj. množinou) $\{w \mid w \text{ má vlastnost } P\}$.

Příklad. Problém rozhodnout, zda daná regulární gramatika reprezentuje konečný jazyk, ztotožníme s jazykem

$$\{\mathcal{G} \mid \mathcal{G} \text{ je regulární gramatika a } L(\mathcal{G}) \text{ je konečný}\}.$$

Definice (str. 109 ve skriptech)

Problém P odpovídající jazyku $L = \{w \mid w \text{ má vlastnost } P\}$ je

- **rozhodnutelný**, právě když L je rekursivní,
- **nerozhodnutelný**, právě když L není rekursivní,
- **částečně rozhodnutelný (semirozhodnutelný)**, právě když L je rekursivně spočetný (r.e.).

Church-Turingova teze: Každý proces, který lze intuitivně nazvat algoritmem, se dá realizovat na úplném Turingově stroji.

Univerzální TM

- každý TM lze zapsat jako číslo

Univerzální TM

- každý TM lze zapsat jako číslo
- z každého čísla lze zrekonstruovat TM a odsimulovat ho

Univerzální TM

- každý TM lze zapsat jako číslo
- z každého čísla lze zrekonstruovat TM a odsimulovat ho

Univerzální Turingův stroj

$\mathcal{U}$ akceptuje $\langle \mathcal{M} \rangle \sharp \langle w \rangle \quad \overset{def}{\Longleftrightarrow} \quad \mathcal{M}$ akceptuje w, tj.

$$L_{\mathcal{U}} = \{ \ \langle \mathcal{M} \rangle \sharp \langle w \rangle \mid \mathcal{M} \text{ akceptuje } w \ \}.$$

Univerzální TM

- každý TM lze zapsat jako číslo
- z každého čísla lze zrekonstruovat TM a odsimulovat ho

Univerzální Turingův stroj

$\mathcal{U}$ akceptuje $\langle\mathcal{M}\rangle\sharp\langle w\rangle \stackrel{def}{\Longleftrightarrow} \mathcal{M}$ akceptuje w, tj.

$$L_{\mathcal{U}} = \{ \langle\mathcal{M}\rangle\sharp\langle w\rangle \mid \mathcal{M} \text{ akceptuje } w \}.$$

Nemůže být úplný:-(

Nerozhodnutelnost

Věta 5.3

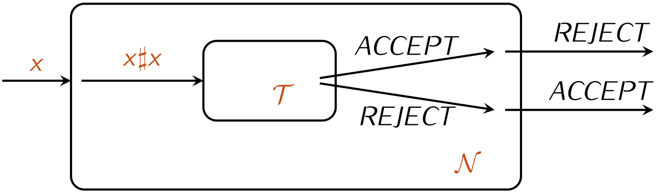
Jazyk $L_{PP} \stackrel{def}{=} \{ \langle \mathcal{M} \rangle \sharp \langle w \rangle \mid \text{ stroj } \mathcal{M} \text{ akceptuje } w \}$ není rekursivní.

Nerozhodnutelnost

Věta 5.3

Jazyk $L_{PP} \stackrel{def}{=} \{ \langle \mathcal{M} \rangle \sharp \langle w \rangle \mid \text{ stroj } \mathcal{M} \text{ akceptuje } w \}$ není rekursivní.

Pro spor předpokládejme, že existuje úplný stroj $\mathcal{T}$ akceptující L_{PP}.



$\mathcal{N}$ akceptuje $x \iff \mathcal{T}$ zamítá $x \sharp x$ (def. $\mathcal{N}$)

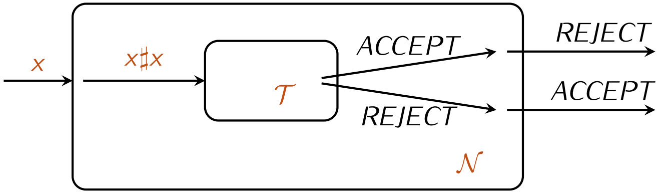
$\iff$ stroj s kódem x zamítá anebo cyklí na x (def. $\mathcal{T}$)

Nerozhodnutelnost

Věta 5.3

Jazyk $L_{PP} \stackrel{def}{=} \{ \langle \mathcal{M} \rangle \sharp \langle w \rangle \mid$ stroj $\mathcal{M}$ akceptuje $w\}$ není rekursivní.

Pro spor předpokládejme, že existuje úplný stroj $\mathcal{T}$ akceptující L_{PP}.



$\mathcal{N}$ akceptuje x $\iff$ $\mathcal{T}$ zamítá $x \sharp x$ (def. $\mathcal{N}$)

$\iff$ stroj s kódem x zamítá anebo cyklí na x (def. $\mathcal{T}$)

Speciálně pro $x = \langle \mathcal{N} \rangle$:

$\mathcal{N}$ akceptuje $\langle \mathcal{N} \rangle$ $\iff$ $\mathcal{T}$ zamítá $\langle \mathcal{N} \rangle \sharp \langle \mathcal{N} \rangle$

$\iff$ stroj s kódem $\langle \mathcal{N} \rangle$ zamítá anebo cyklí na $\langle \mathcal{N} \rangle$

$\iff$ $\mathcal{N}$ neakceptuje $\langle \mathcal{N} \rangle$

Diagonalizace: Cantor

$$
\begin{array}{lcl}
s_1 & = & \mathbf{0}\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \ldots \\
s_2 & = & 1\ \mathbf{1}\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ \ldots \\
s_3 & = & 0\ 1\ \mathbf{0}\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ \ldots \\
s_4 & = & 1\ 0\ 1\ \mathbf{0}\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ \ldots \\
s_5 & = & 1\ 1\ 0\ 1\ \mathbf{0}\ 1\ 1\ 0\ 1\ 0\ 1\ \ldots \\
s_6 & = & 0\ 0\ 1\ 1\ 0\ \mathbf{1}\ 1\ 0\ 1\ 1\ 0\ \ldots \\
s_7 & = & 1\ 0\ 0\ 0\ 1\ 0\ \mathbf{0}\ 0\ 1\ 0\ 0\ \ldots \\
s_8 & = & 0\ 0\ 1\ 1\ 0\ 0\ 1\ \mathbf{1}\ 0\ 0\ 1\ \ldots \\
s_9 & = & 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ \mathbf{1}\ 1\ 0\ \ldots \\
s_{10} & = & 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ \mathbf{0}\ 1\ \ldots \\
s_{11} & = & 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ \mathbf{0}\ \ldots \\
\vdots & & \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \ddots
\end{array}
$$

$$
s = 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ \ldots
$$

Diagonalizace: Cantor

$$
\begin{array}{ll}
s_1 & = \color{red}{0}\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ \ldots \\
s_2 & = 1\ \color{red}{1}\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ \ldots \\
s_3 & = 0\ 1\ \color{red}{0}\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ \ldots \\
s_4 & = 1\ 0\ 1\ \color{red}{0}\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ \ldots \\
s_5 & = 1\ 1\ 0\ 1\ \color{red}{0}\ 1\ 1\ 0\ 1\ 0\ 1\ \ldots \\
s_6 & = 0\ 0\ 1\ 1\ 0\ \color{red}{1}\ 1\ 0\ 1\ 1\ 0\ \ldots \\
s_7 & = 1\ 0\ 0\ 0\ 1\ 0\ \color{red}{0}\ 0\ 1\ 0\ 0\ \ldots \\
s_8 & = 0\ 0\ 1\ 1\ 0\ 0\ 1\ \color{red}{1}\ 0\ 0\ 1\ \ldots \\
s_9 & = 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ \color{red}{1}\ 1\ 0\ \ldots \\
s_{10} & = 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ \color{red}{0}\ 1\ \ldots \\
s_{11} & = 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ \color{red}{0}\ \ldots \\
\vdots & \quad \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \ddots
\end{array}
$$

$$
s = \color{blue}{1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 1}\ \ldots
$$

$f : S \to \mathcal{P}(S)$

$$
\begin{array}{ll}
f(1) & = \{ \} \\
f(2) & = \{ 1, \color{red}{2,} 3, 4, 5, 6, 7, 8, 9, 10, 11, \ldots \} \\
f(3) & = \{ 2, 4, 6, 8, 10, \ldots \} \\
f(4) & = \{ 1, 3, 5, 7, 9, 11, \ldots \} \\
f(5) & = \{ 1, 2, 4, 6, 7, 9, 11, \ldots \} \\
f(6) & = \{ 3, 4, \color{red}{6,} 7, 9, 10, \ldots \} \\
f(7) & = \{ 1, 5, 9, \ldots \} \\
f(8) & = \{ 3, 4, 7, \color{red}{8,} 11, \ldots \} \\
f(9) & = \{ 1, 2, 5, 6, \color{red}{9,} 10, \ldots \} \\
f(10) & = \{ 1, 2, 4, 5, 6, 9, 11, \ldots \} \\
f(11) & = \{ 1, 2, 4, 6, 9, \ldots \} \\
\vdots & \quad \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \vdots\ \ddots
\end{array}
$$

$$
T = \{ \color{blue}{1, 3, 4, 5, 7, 10, 11}, \ldots \}
$$

$C = \{ s \in S \mid s \notin f(s) \}$

Diagonalizace: Russell

$R = \{x \mid x \notin x\}$

Diagonalizace: Russell

$R = \{x \mid x \notin x\}$

Pak $R \in R \iff R \notin R$

Diagonalizace: Turing

	w_ε	w_0	w_1	w_{00}	w_{01}	w_{10}	w_{11}	w_{000}	w_{001}	$\cdots$
$\mathcal{M}_\varepsilon$	1	0	1	1	0	1	0	1	1	
$\mathcal{M}_0$	0	1	1	1	0	1	0	0	1	
$\mathcal{M}_1$	1	1	1	0	1	1	0	1	0	
$\mathcal{M}_{00}$	0	0	0	0	0	1	0	1	1	
$\mathcal{M}_{01}$	1	0	0	1	0	0	1	1	1	$\cdots$
$\mathcal{M}_{10}$	0	0	0	1	1	1	0	1	1	
$\mathcal{M}_{11}$	1	1	0	1	0	1	0	1	0	
$\mathcal{M}_{000}$	0	1	1	0	0	1	1	1	1	
$\mathcal{M}_{001}$	1	1	1	1	1	0	0	1	0	
$\vdots$					$\vdots$					$\ddots$
$\mathcal{N}$	0	0	0	1	1	0	1	0	1	

Diagonalizace: Turing

	w_ε	w_0	w_1	w_{00}	w_{01}	w_{10}	w_{11}	w_{000}	w_{001}	$\cdots$
$\mathcal{M}_\varepsilon$	1	0	1	1	0	1	0	1	1	
$\mathcal{M}_0$	0	1	1	1	0	1	0	0	1	
$\mathcal{M}_1$	1	1	1	0	1	1	0	1	0	
$\mathcal{M}_{00}$	0	0	0	0	0	1	0	1	1	
$\mathcal{M}_{01}$	1	0	0	1	0	0	1	1	1	$\cdots$
$\mathcal{M}_{10}$	0	0	0	1	1	1	0	1	1	
$\mathcal{M}_{11}$	1	1	0	1	0	1	0	1	0	
$\mathcal{M}_{000}$	0	1	1	0	0	1	1	1	1	
$\mathcal{M}_{001}$	1	1	1	1	1	0	0	1	0	
$\vdots$					$\vdots$					$\ddots$
$\mathcal{N}$	0	0	0	1	1	0	1	0	1	

Věta 5.5

Jazyk $co{-}L_{PP} \stackrel{def}{=} \{\langle\mathcal{M}\rangle\sharp\langle w\rangle \mid \mathcal{M} \text{ neakceptuje } w\}$ není r.e.

Důkaz. L_{PP} je r.e. Pokud $co{-}L_{PP}$ je také r.e., pak jsou oba rekursivní, což je spor. □

Problém zastavení

Věta 5.6

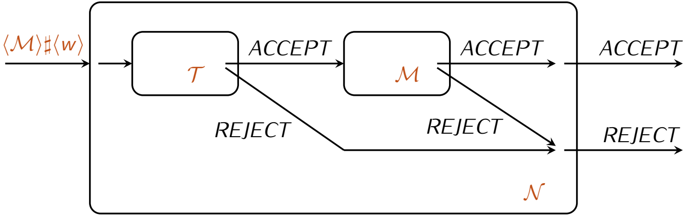
Problém zastavení pro Turingovy stroje není rozhodnutelný, tj.
$L_{PZ} \stackrel{def}{=} \{\langle\mathcal{M}\rangle\sharp\langle w\rangle \mid \text{ výpočet } \mathcal{M} \text{ na } w \text{ je konečný } \}$ není rekursivní.

Problém zastavení

Věta 5.6

Problém zastavení pro Turingovy stroje není rozhodnutelný, tj.
$L_{PZ} \stackrel{def}{=} \{\langle\mathcal{M}\rangle\sharp\langle w\rangle \mid \text{ výpočet } \mathcal{M} \text{ na } w \text{ je konečný } \}$ není rekursivní.

Pro spor předpokládejme, že existuje úplný TM $\mathcal{T}$ akceptující L_{PZ}.
Pomocí $\mathcal{T}$ můžeme sestrojit úplný TM $\mathcal{N}$ rozhodující jazyk L_{PP}, spor.

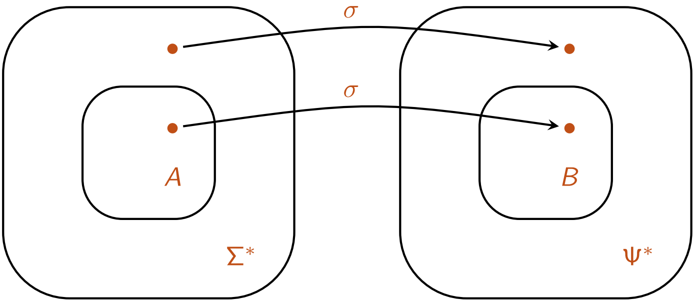


Redukce

Definice 5.7

Redukce jazyka $L_A \subseteq \Sigma^*$ na jazyk $L_B \subseteq \Psi^*$ je rekursivní funkce $\sigma : \Sigma^* \to \Psi^*$ taková, že

$$w \in L_A \iff \sigma(w) \in L_B \ .$$



Redukce

Věta 5.8

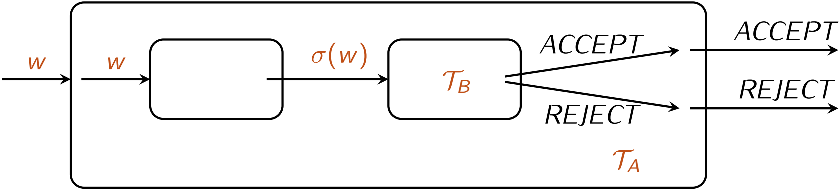
Nechť existuje redukce L_A na L_B (píšeme $L_A \leq L_B$).

(*i*) Není-li L_A r.e., pak ani L_B není r.e.

(*ii*) Není-li L_A rekursivní, pak ani L_B není rekursivní.

(*i*) Je-li L_B r.e., pak L_A je r.e.

(*ii*) Je-li L_B rekursivní, pak L_A je rekursivní.



Redukce

Věta 5.8

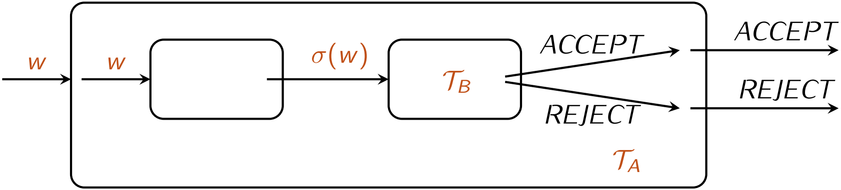
Nechť existuje redukce L_A na L_B (píšeme $L_A \leq L_B$).

(*i*) Není-li L_A r.e., pak ani L_B není r.e.

(*ii*) Není-li L_A rekursivní, pak ani L_B není rekursivní.

(*i*) Je-li L_B r.e., pak L_A je r.e.

(*ii*) Je-li L_B rekursivní, pak L_A je rekursivní.



Příklad: $L_{PP} \leq L_{PZ}$ pomocí $\langle \mathcal{M} \rangle \sharp \langle w \rangle \mapsto \langle \mathcal{M}' \rangle \sharp \langle w \rangle$,
kde $\mathcal{M}'$ je jako $\mathcal{M}$, ale když $\mathcal{M}$ zamítá, přejde $\mathcal{M}'$ do cyklení