# Redukce – příklady ve cvičeních (řešení)

**12.1** Rozhodněte, zda platí následující implikace. Své rozhodnutí zdůvodněte.

a) $A \leq_m B \Rightarrow \overline{A} \leq_m \overline{B}$

b) $A \leq_m B$ a *B* je regulární ⇒ *A* je regulární.

c) *A* je rekurzivně spočetná a $\overline{A} \leq_m A \Rightarrow A$ je rekurzivní.

d) *A* je rekurzivně spočetná a $A \leq_m \overline{A} \Rightarrow A$ je rekurzivní.

e) $A \leq_m B$ a *A* je rekurzivní ⇒ *B* je rekurzivní

f) *A* je rekurzivně spočetná ⇒ $A \leq_m HALT$

Řešení:

a) **Platí.** Buď $A \subseteq \Sigma^*, B \subseteq \Gamma^*$. Zopakujme definici m-redukce:
$A \leq_m B$ právě tehdy, když existuje totálně vyčíslitelná funkce $f: \Sigma^* \to \Gamma^*$ taková, že
$(*)\ w \in A \Leftrightarrow f(w) \in B$
Vztah (*) můžeme upravit: $w \notin A \Leftrightarrow f(w) \notin B$, resp.
$(**)\ w \in \overline{A} \Leftrightarrow f(w) \in \overline{B}$
Vztah (**) však odpovídá definici m-redukce $\overline{A} \leq_m \overline{B}$, což bylo dokázat.

b) **Neplatí.** Uvedeme protipříklad. Nechť
$A = \{w.w \mid w \in \{a,b\}^*\}$,
$B \subseteq \{0,1\}^*, B = \{0\}$.
Jazyk *A* je rekurzivní, jazyk *B* regulární. Zkonstruujme funkci $f: \{a,b\}^* \to \{0,1\}$ pro redukci $A \leq_m B$:
$f(x) = 0$, je-li $x$ tvaru $w.w$, kde $w \in \{a,b\}^*$,
$f(x) = 1$ jinak.
Tato funkce je totálně vyčíslitelná, protože pro ni existuje Turingův stroj *M*, který rozhoduje jazyk *A*, tudíž funkci $f$ lze na **jakémkoliv** vstupu z $\{a,b\}^*$ algoritmicky spočítat. Našli jsme tedy neregulární jazyk *A* a regulární jazyk *B*, pro něž jsme zkonstruovali redukci $A \leq_m B$. Spor s tvrzením v zadání.

c) **Platí.** Nejprve uveďme jednu z vlastností m-redukce:
(*) Pokud *Y* rekurzivně spočetný a platí $X \leq_m Y$, pak *X* je též rekurzivně spočetný.
Porovnejme vztah (*) s předpoklady věty v zadání: *A* je rekurzivně spočetná a $\overline{A} \leq_m A$. Dle vztahu (*) tedy musí platit, že i jazyk $\overline{A}$ je rekurzivně spočetný. Ukázali jsme, že oba jazyky $A, \overline{A}$ jsou rekurzivně spočetné. Z toho však vyplývá, že jazyk *A* je rekurzivní.

d) **Platí.** Ze vztahu $\overline{A} \leq_m A$ vyplývá i vztah $A \leq_m \overline{A}$ (viz definice m-redukce). Dále již postupujeme analogicky jako v případě c).

e) **Neplatí.** Uvažujme následující dva jazyky:
$A = \emptyset, A \subseteq \Sigma^*$
$B = HALT$, tj. jazyk nad abecedou $\{0,1,\#\}$ obsahující kódy $\langle M, w\rangle$, kde *M* je Turingův stroj a výpočet *M* na slově *w* je konečný. Sestrojme funkci *f,* která bude pro jakýkoliv vstup vracet slovo nad abecedou $\{0,1,\#\}$ nepatřící do *B* (tedy nebude to smysluplný kód tvaru $\langle M, w\rangle$). Jde o totálně vyčíslitelnou funkci, přičemž $x \in \Sigma^*$ platí $x \notin A \Leftrightarrow f(x) \notin B$. Tedy $A \leq_m B$, přičemž *A* je regulární, tedy i rekurzivní jazyk, naopak *B* je pouze rekurzivně spočetný jazyk, nikoliv rekurzivní.

**f)** **Platí.** Jestliže *A* je rekurzivně spočetný jazyk, existuje pro něj Turingův stroj *M*, který je akceptuje. Je-li $A \subseteq \Sigma^*$, pak na slovech $w \in A$ Turingův stroj *M* akceptuje, na slovech $w' \notin A$ zamítá či cyklí. Sestrojíme funkci $f: \Sigma^* \rightarrow \{0, 1, \#\}^*$ takto:

$f(w) = \langle N, w \rangle$, kde *N* vznikne z *M* tak, že přechody do stavu $q_{rej}$ upravíme tak, aby v *N* cyklily.

Taková funkce je určitě totálně vyčíslitelná. Výstup funkce $f(w)$, tj. kód $\langle N, w \rangle$ přesměrujeme do Turingova stroje pro jazyk *HALT*. Jistě platí:
$\langle M, w \rangle \in A \Leftrightarrow \langle N, w \rangle \in HALT$
$x \in A \Leftrightarrow f(x) \in HALT$
Našli jsme tedy redukci $A \leq_m HALT$.

**12.2** Je dán jazyk $A = \{\langle M \rangle \mid$ výpočet TM *M* na slově $\varepsilon$ je konečný$\}$.

- Dokažte, že *A* není rekurzivní. (Návod: najděte redukci problému zastavení na *A*.)
- Je jazyk *A* rekurzivně spočetný?
- Je komplement jazyka *A* rekurzivně spočetný?

<u>Řešení:</u>

**Důkaz, že jazyk *A* není rekurzivní**

Zopakujme si nejdříve definici jazyku *HALT*. Jedná se o jazyk nad abecedou $\{0, 1, \#\}$ obsahující kódy $\langle M, w \rangle$, kde *M* je Turingův stroj a výpočet *M* na slově *w* je konečný. Zkonstruujme funkci $f$: $\{0, 1, \#\}^* \rightarrow \{0, 1, \#\}^*$, která zobrazí slovo $x = \langle M, w \rangle$ na jiné slovo $f(x) = \langle N, w \rangle$ tak, že výpočet *M* na *w* je konečný právě tehdy, když výpočet *N* na $\varepsilon$ je konečný.

$f(x) = \langle T_\infty \rangle$, pokud $x \neq \langle M, w \rangle$, kde $T_\infty$ je TM, který pro každý vstup (tedy i prázdné slovo) cyklí.
$f(x) = \langle N, u \rangle$, pokud $x = \langle M, w \rangle$, kde stroj *N* pro libovolný vstup *u* sestrojíme takto:

1. Spustíme simulaci stroje *M* na slově *w.*
2. Pokud *M* zastaví svůj výpočet, akceptujeme.

Funkce *f* je určitě vyčíslitelná.

- V případě, že vstupem funkce není kód $\langle M, w \rangle$ Turingova stroje a vstupu, výsledkem je kód stroje $T_\infty$, který pro každý vstup cyklí. Takový stroj jsme jistě schopni vymyslet.
- V opačném případě zkonstruujeme Turingův stroj *N*, který při své činnosti simuluje činnost stroje *M* na vstupu *w*. I takový stroj sestrojíme jednoduše, stejně tak si poradíme s výstupem v případě, že *M* na vstupu *w* zastaví. Pokud by výpočet *M* na *w* cyklil, $\langle M, w \rangle \notin HALT$, tak bude cyklit výpočet *N* na libovolném slově *u*, tedy i $\varepsilon$, tj. $\langle N, u \rangle \notin A$.

Funkce *f* je též totální, protože je definována pro každý vstup. Jejím výsledkem je vždy řetězec nad abecedou $\{0, 1, \#\}$ reprezentující buď pouze kód TM nebo kód dvojice TM, vstup. Platí:

$\langle M, w \rangle \in HALT \Leftrightarrow \langle N, u \rangle \in A$ (Výpočet *M* na w je konečný $\Leftrightarrow$ výpočet *N* na libovolném slově *u*, tedy i $\varepsilon$, je konečný)
$x \in HALT \Leftrightarrow f(x) \in A$
$HALT \leq_m A$

Protože jazyk *HALT* není rekurzivní, není ani jazyk *A* rekurzivní.

**Důkaz, že jazyk *A* je rekurzivně spočetný**

Důkaz faktu, že jazyk *A* je rekurzivně spočetný, provedeme redukcí $A \leq_m ACC$. Protože je jazyk *ACC* rekurzivně spočetný, jistě je i jazyk *A* rekurzivně spočetný. Zkonstruujme funkci $f'$: $\{0, 1, \#\}^* \rightarrow \{0, 1, \#\}^*$, která zobrazí slovo $x = \langle M \rangle$ na slovo $f'(x) = \langle N, \varepsilon \rangle$, kde *M, N* jsou Turingovy stroje, pro které platí, že výpočet *M* na prázdném slově $\varepsilon$ zastaví právě tehdy, když *N* akceptuje $\varepsilon$.

$f'(x) = x$, není-li *x* kódem Turingova stroje.

$f'(x) = \langle N, \varepsilon \rangle$, je-li $x = \langle M \rangle$, kde *N* zkonstruujeme následujícím způsobem:

1. Univerzálnímu TM *U* dáme na vstup dvojici $\langle M, \varepsilon \rangle$.
2. Pokud *U* akceptuje, stroj *N* také akceptuje.
3. Pokud *U* zamítá, stroj *N* akceptuje.

Funkce *f'* je jistě vyčíslitelná. TM *N* při svém výpočtu využívá univerzálního TM *U*, který jsme schopni zkonstruovat. Funkce *f'* je totální, je definovaná jak pro špatné vstupy (*x* není kódem TM), tak pro korektní vstupy (*x* je kódem TM). Platí:

$\langle M \rangle \in A \Leftrightarrow \langle N, \varepsilon \rangle \in ACC$
$x \in A \Leftrightarrow f'(x) \in ACC$
$A \leq_m ACC$

Jazyk *A* je tedy rekurzivně spočetný.

**Důkaz, že jazyk co-*A* není ani rekurzivně spočetný**

Sporem předpokládejme, že jazyk co-*A* je rekurzivně spočetný. Protože i jazyk *A* je rekurzivně spočetný, z faktu, že oba jazyky jsou rekurzivně spočetné, vyplývá, že jazyk *A* je rekurzivní. To je však spor s naším důkazem, že *A* není rekurzivní. Proto jazyk co-*A* není ani rekurzivně spočetný.

**12.3** Nalezněte řešení následujícího Postova systému:

$$\left\{\left[\frac{aa}{a}\right], \left[\frac{ab}{abab}\right], \left[\frac{b}{a}\right], \left[\frac{aba}{b}\right]\right\}$$

<u>Řešení:</u>

1343421343222

$$\left[\frac{aa}{a}\right] \cdot \left[\frac{b}{a}\right] \cdot \left[\frac{aba}{b}\right] \cdot \left[\frac{b}{a}\right] \cdot \left[\frac{aba}{b}\right] \cdot \left[\frac{ab}{abab}\right] \cdot \left[\frac{aa}{a}\right] \cdot \left[\frac{b}{a}\right] \cdot \left[\frac{aba}{b}\right] \cdot \left[\frac{b}{a}\right] \cdot \left[\frac{ab}{abab}\right] \cdot \left[\frac{ab}{abab}\right] \cdot \left[\frac{ab}{abab}\right]$$

2243311

$$\left[\frac{ab}{abab}\right] \cdot \left[\frac{ab}{abab}\right] \cdot \left[\frac{aba}{b}\right] \cdot \left[\frac{b}{a}\right] \cdot \left[\frac{b}{a}\right] \cdot \left[\frac{aa}{a}\right] \cdot \left[\frac{aa}{a}\right]$$

**12.5** Ukažte, že problém ekvivalence dvou Turingových strojů

$$EQ = \{\langle M_1, M_2 \rangle \mid M_1, M_2 \text{ jsou } TM \wedge L(M_1) = L(M_2)\}$$

je nerozhodnutelný.

<u>Řešení:</u>

Z přednášky víme, že jazyk $NONEMPTY = \{\langle M \rangle \mid M \text{ je } TM \wedge L(M) \neq \emptyset\}$ není rekurzivní. Uvažujme nyní $EMPTY = \{\langle M \rangle \mid M \text{ je } TM \wedge L(M) = \emptyset\}$. Naším plánem je ukázat, že není rekurzivní, a následně jej použít v redukci $EMPTY \leq_m EQ$, čímž bychom dokázali, že i jazyk $EQ$ není rekurzivní.

Až na řetězce $\langle y \rangle$, které nekódují žádný TM, je jazyk $EMPTY$ doplňkem jazyka $NONEMPTY$. Sjednotíme-li tedy množiny $EMPTY$ a $B = \{\langle y \rangle \mid y \text{ není kódem } TM\}$, měli bychom dostat jazyk, který není rekurzivní. Pokud by byl, musel by být díky uzávěrovým vlastnostem rekurzivní i $NONEMPTY$. Tedy

(*) $EMPTY \cup B$ není rekurzivní.

Jazyk $EMPTY$ však také není rekurzivní. Není totiž složité najít Turingův stroj, který by rozhodoval jazyk *B*, tedy *B* je rekurzivní. Jazyk $EMPTY$ tím pádem není rekurzivní – pokud by byl, bylo by i sjednocení $EMPTY \cup B$ díky uzávěrovým vlastnostem rekurzivní, což je spor s výše uvedeným závěrem (*).

Nyní přistupme k popsání redukce $EMPTY \leq_m EQ$. Sestrojme funkci $f: \{0,1,\#\}^* \to \{0,1,\#\}^*$, která zobrazí slovo $x = \langle M \rangle$ na slovo $f(x) = \langle M, M_1 \rangle$ tak, že $x \in EMPTY \Leftrightarrow f(x) \in EQ$, přičemž $L(M_1) = \emptyset$. Předpokládejme, že existuje TM *R* rozhodující jazyk $EQ$.

$f(x) = \langle M, M_1 \rangle$, kde $M_1$ je TM zamítající jakýkoliv vstup.

Nyní ověřme:

$\langle M \rangle \in EMPTY \Leftrightarrow L(M) = \emptyset \Leftrightarrow \langle M, M_1 \rangle \in EQ$
$x \in EMPTY \Leftrightarrow f(x) \in EQ$
$EMPTY \leq_m EQ$

Není-li *x* kódem Turingova stroje, platí $L(x) = \emptyset$, tj. $x \in EMPTY$. TM *R* tento kód porovnává s kódem TM $M_1$, pro který taktéž platí $L(M_1) = \emptyset$. $R$ tedy přijímá. V opačném případě, kdy $x = \langle M \rangle$, kde $M$ je nějaký TM, připravíme vstup $\langle M, M_1 \rangle$ pro TM *R* rozhodující jazyk $EQ$. *R* přijímá pouze tehdy, když $L(M) = L(M_1) = \emptyset$, tedy když $f(x) = \langle M, M_1 \rangle \in EQ$.

Funkce *f* je totální, jelikož každému vstupu *x* přiřazuje kód $\langle M, M_1 \rangle$. Je i vyčíslitelná, jelikož při své práci pouze simuluje činnost TM *R*. Protože však jazyk $EMPTY$ není rekurzivní, nemůže být rekurzivní ani jazyk $EQ$.