J003 – Fundamental Concepts of Computer Science and Some Surprising Discoveries: 2. sada

Karel Kubíček

26. května 2015

Úloha 1

K důkazu použijeme totálně vyčíslitelnou funkci, která provede redukci slov z jazyka L_H do slov jazyka L_U.

$$f(w) = \begin{cases} \langle \mathcal{T}_{\text{inf}} \rangle, & \text{jestliže } w \text{ není kódem žádné dvojice skládající se z TS a slova,} \\ \langle \mathcal{T}_{\mathcal{M},u} \rangle, & \text{jestliže } w = \langle \mathcal{M}, u \rangle \text{ pro nějaký TS } \mathcal{M} \text{ a nějaké slovo } u, \end{cases}$$

Kde $\mathcal{T}_{\text{inf}}$ je TS, který cyklí. Zato $\mathcal{T}_{\mathcal{M},u}$ je TS, který simuluje TS M na slově u a to vždy akceptuje.

Funkce f je totálně vyčíslitelná, jedná se tedy o Turingovu redukci. Musíme ještě dokázat, že redukce zachovává příslušnost slov v jazyce.

„Pokud $w \in L_H \implies f(w) \in L_U$“: jelikož $w \in L_H$, pak jistě simulovaný stroj zastaví, a tedy je slovo akceptováno.

„Pokud $w \notin L_H \implies f(w) \notin L_U$“: to, že $w \notin L_H$ znamená, že simulovaný stroj cyklí (buď w není kódem dvojice TM a slova, pak cyklí strom $\mathcal{T}_{\text{inf}}$, nebo je $w = \mathcal{T}_{\mathcal{M},u}$, a tedy $\mathcal{T}$ cyklí nad u). Pak výsledný stroj nedojde do akceptujícího stavu, a tedy $f(w) \notin L_U$.

Úloha 2

TODO

Úloha 3

(a) Řešení je obdobné, jako u úlohy 1, jen v redukci předáme slovo 001 tak, aby se simulovaný stroj pouštěl vždy na něm.

(b) Redukce funguje následovně.

$$f(w) = \begin{cases} \langle \mathcal{T}_{\emptyset} \# \mathcal{T}_{\Sigma^*} \rangle, & \text{jestliže } w \text{ není kódem žádné dvojice skládající se z TS a slova,} \\ \langle \mathcal{T}_u \# (\mathcal{T}_u \cap M) \rangle, & \text{jestliže } w = \langle \mathcal{M}, u \rangle \text{ pro nějaký TS } \mathcal{M} \text{ a nějaké slovo } u, \end{cases}$$

Kde první případ generuje TS akceptující prázdný jazyk a TS akceptující všechna slova, které jsou odlišné, a tedy je takový vstup zamítnut.

V druhém případě k zadané dvojici M, u vytvoříme v redukci stroj $\mathcal{T}_u$, který akceptuje pouze slovo u. Vstupem pro TS jazyka L_{EQ} je $\mathcal{T}_u$, oddělovač # a průnik $\mathcal{T}_u \cap M$, přičemž průnik lze snadno realizovat pomocí spuštění obou TS a akceptováním pouze v případě, že oba TS skončily v akceptujícím stavu.

Úloha 5

Důkaz provedeme redukováním $L_U \leq_R L_{all}$, přičemž víme, že $L_U \notin \mathcal{L}_R$.

Redukční funkce bude pro vstup $u = \text{Kod}(M)\#w$ následovný Turingův stroj (pokud u nekóduje TS # a slovo, pak vrací TS prázdného jazyka).

TS se podívá na vstupní slovo v. Pokud $v \neq u$, pak jej akceptuje. Pokud $v = u$, pak spustí M na v.

Úloha 7

Tvrzení dokážeme konstrukcí daného TS. V obou případech se bude jednat o třípáskové stroje, kde první páska obsahuje vstup a na dalších 2 probíhá výpočet.

(a) První páska obsahuje vstup, po každé iteraci se na této pásce posouvá čtecí hlava o jedno pole dále a pokud narazí na pravou zarážku, tak ukončí výpočet a výstup překopíruje z aktuální pásky na první pracovní pásku.

Zbylé pásky před začátkem výpočtu obsahují po 1 symbolu 0. Iterace překopíruje za konec jedné pásky tolik 0, kolik je na druhé pásce. V další iteraci se zapisuje na druhou pásku.

Cena výpočtu je v $\mathcal{O}(2^n \cdot 4)$. Násobek 4 je dán nutností přesunu z konce pásky na začátek před započtením kopírování a závěrečným překopírováním výstupu. To je v $\mathcal{O}(e(n))$.

(b) Výpočet fibonacciho čísel probíhá obdobně jako výpočet mocniny 2. Změna je v tom, že na začátku inicializujeme jednu pásku na 0 a druhou na 1.