Úkol 1

Příklad 1

Navrhněte Turingův stroj, který počítá součet dvou čísel v desítkové soustavě.

- Vstup je ve tvaru $\triangle\alpha\#\beta\triangle^\omega$, kde α a β jsou čísla v desítkové soustavě, $\alpha, \beta > 0$.
- Po zastavení Turingova stroje první páska obsahuje $\triangle\alpha\#\beta\#\gamma\triangle^\omega$, kde $\gamma = \alpha + \beta$.
- V případě, že navrhnete vícepáskový stroj, obsah dalších pásek může být libovolný.
- Určete horní odhad časové a prostorové složitosti tohoto stroje.

Řešení 1

Sestrojíme čtyř páskový Turingův stroj. Obsah jednotlivých pásek bude následovný.

- První páska bude obsahovat zadání ve tvaru $\triangle\alpha\#\beta\triangle^\omega$ a následně výsledek ve tvaru $\triangle\alpha\#\beta\#\gamma\triangle^\omega$.
- Druhá páska bude obsahovat zkopírovanou hodnotu druhého operandu β ve tvaru $\triangle\beta\triangle^\omega$.
- Třetí páska bude obsahovat pouze jednu buňku udávající informaci o přetečení. Páska je inicializovaná na hodnotu $\triangle 0\triangle^\omega$.
- Čtvrtá páska bude obsahovat reverzi průběžného výsledku sčítání. Výsledná hodnota bude na pásce ve tvaru $\triangle\gamma^R\triangle^\omega$, kde γ^R je reverzní řetězec k řetězci γ.

Nechť je dálka obsahu vstupu n. V nejhorším případě bude i délka největšího operandu $n - 2 \sim n$. Výpočet Turingova stroje probíhá následovně. Poznamenejme, že logika pro součet dvou čísel 0–9 a carry bitu je již zakódována ve struktuře stroje.

1. Posuneme hlavu na 1. pásce až na MSB[1] β. $O_{time}(n + 2)$
2. Z 1. pásky se zkopíruje operand β na 2 pásku. (Kopírování sestává z 3 operací: zápisu z pod hlavy na 1. pásce na 2. pásku, posunu hlavy na 1. pásce doprava a posunu hlavy na 2. pásce doprava). $O_{time}(3n)$
3. Posuneme hlavu na 1. pásce z koncového znaku $\triangle$ na LSD[2] operandu α. $O_{time}(n + 2)$
4. Přesuneme hlavu na 2. pásce z koncového znaku $\triangle$ na LSD operandu β. $O_{time}(1)$
5. Přesuneme hlavu na 3. pásce z počátečního znaku $\triangle$ doprava na hodnotu 0. $O_{time}(1)$
6. Přesuneme čtecí hlavu na 4. pásce z počátečního znaku $\triangle$ doprava. $O_{time}(1)$
7. Sečteme hodnoty pod hlavami na 1. (α), 2. (β) a 3. (carry) pásce. Pokud je čtecí hlava na první, nebo druhé pásce na znaku $\triangle$, pak se tento znak interpretuje jako 0. Pokud je výsledek součtu z intervalu $\langle 0, 9\rangle$, pak se zapíše na 4. pásku a na 3. pásku se zapíše 0. Pokud je výsledek z intervalu $\langle 10, 19\rangle$, pak se zapíše hodnota na místě jednotek na 4. pásku a na 3. pásku se zapíše hodnota 1. $O_{time}(2)$
8. Pokud je pod hlavou na 1. pásce znak různý od $\triangle$, pak se posune hlava doleva. $O_{time}(1)$
9. Pokud je pod hlavou na 2. pásce znak různý od $\triangle$, pak se posune hlava doleva. $O_{time}(1)$
10. Na 4. pásce se posune hlava doprava. $O_{time}(1)$
11. Pokud se pod alespoň pod jednou z hlav z 1., nebo 2. pásky vyskytuje znak různý od $\triangle$, tak se výpočet vrací na bod 7, jinak pokračuje.
12. Pokud se pod hlavou na 3. pásce nachází 1, tak se překopíruje na 4. pásku a potom se posune hlava na pásce 4 doprava. $O_{time}(2)$

[1] Most Significant Digit (nejlevější číslice)
[2] Least Significant Digit (nejpravější číslici)

13. Posuň hlavu na 1. pásce na první pravý $\triangle$. $O_{time}(n)$
14. Zapiš na 1. pásku znak # a posuň hlavu doprava. $O_{time}(2)$
15. Posuň hlavu na 4. pásce doleva (na MSD výsledku). $O_{time}(1)$
16. Postupně obráceně kopíruj reverzní výsledek z 4. pásky na konec 1. pásky. (Kopírování vyžaduje tři operace.) $O(3n)$
17. Výsledek sčítání je na 1. pásce ve tvaru $\triangle\alpha\#\beta\#\gamma\triangle^{\omega}$.

Horní odhad časové složitosti navrženého Turingova stroje je $O_{time}\big((n+2)+(3n)+(n+2)+(3)+n\cdot(5)+(2)+(n)+(3)+(3n)\big) = O_{time}(n)$.

Horní odhad prostorové složitosti navrženého Turingova stroje je $O_{space}\big((2+2n+1)+(2+n)+(2)+(2+n+1)\big) = O_{space}(n)$.

Příklad 2

Navrhněte RAM program, který pro vstupní vektor $I = (n_1, n_2)$ vypočítá hodnotu n_1 *mod* n_2 (předpokládejme, že $n_1, n2 > 0$). Po provedení instrukce *HALT* bude v registru r_0 číslo $a = n_1$ *mod* n_2. (Pozn: Není třeba implementovat optimální algoritmus.)

- Analyzujte uniformní (jednotkovou) časovou a prostorovou složitost tohoto RAM programu a uveďte horní odhady.
- Analyzujte logaritmickou časovou a prostorovou složitost tohoto RAM programu a uveďte horní odhady.

Řešení 2

RAM program Π, který prování operaci n_1 *mod* n_2 na vstupním vektoru $I = (n_1, n_2)$ vypadá následovně.

```
1: READ   1
2: STORE  1
3: READ   0
4: SUB    r1
5: JPOS   4
6: JZERO  4
7: ADD    r1
8: HALT
```

Uniformní časová složitost stroje je dána počtem vykonaných iterací, kterých je v nejhorším případě n_1 (pro $n_2 = 1$). Při jednotkové ceně instrukce je $O^{uni}_{time}(2^n)$. Uniformní prostorová složitost stroje je dána délkou vstupu a počtem použitých registrů, $O^{uni}_{space}(4) = O^{uni}_{space}(1)$.

Při logaritmické časové složitosti je cena instrukce $log_2(2^n)$. Výsledná logaritmická časová složitost je $O^{log}_{time}(2^n \cdot n)$. Logaritmická prostorová složitost stroje je dána také počtem bitů potřebných k uložení hodnoty, tedy $O^{log}_{space}(4 * n) = O^{log}_{space}(n)$.

Příklad 3

Nechť L je libovolný regulární jazyk. Určete funkce $f(n)$ a $g(n)$ takové, že $L \in DTIME(f(n))$ a $L \in DSPACE(g(n))$. Svoje tvrzení dokažte.

Řešení 3

Pro libovolný jazyk $L \in \mathcal{L}_3$ platí, že existuje Turingův stroj M, který přijímá přechodem do koncového stavu, pro který platí, $L \equiv L(M)$ a jehož časová a prostorová složitost je lineární. Tedy $L \in DTIME(n)$ a $L \in DSPACE(n)$.

Důkaz. Pro každý $L \in \mathcal{L}_3$ exituje deterministický konečný automat A ve tvaru $A = (Q_A, \Sigma_A, \delta_A, i_A, F_A)$, který daný jazyk přijímá. Q_A je konečná množina stavů automatu, Σ_A je konečná vstupní abeceda automatu, δ_A je přechodová funkce tvaru $\delta_A : Q_A \times \Sigma \longrightarrow Q$, dále $i_A \in Q$ je počáteční stav a $F_A \subseteq Q_A$ množina koncových stavů. Pro takovýto automat A můžeme sestrojit jednopáskový Turingův stroj M ve tvaru $M = (Q_M, \Sigma_M, \Gamma_M, \delta_M, i_M, f_M)$, který přijímá jazyk L, kde Q_M je konečná množina stavů, $\Sigma_M \subset \Gamma_M$ je vstupní abeceda, Γ_M je pásková abeceda, δ_M je přechodová funkce ve tvaru $\delta_M : Q_M \times \Gamma \longrightarrow Q \times (\Gamma \cup \{R, L\})$, kde R je posun hlavy doprava a L je posun hlavy doleva, $i_M \in Q_M$ je počáteční stav a $f_M \in Q_M$ je koncový stav.

Turingův stroj $M = (Q_M, \Sigma_M, \Gamma_M, \delta_M, i_M, f_M)$ je z automatu $A = (Q_A, \Sigma_A, \delta_A, i_A, F_A)$ zkonstruován následovně.

- $Q_M = Q_A \cup \{f_M, i_M\}$, kde $f_M, i_M \notin Q_A$
- $\Sigma_M = \Sigma_A$
- $\Gamma_M = \Sigma_A \cup \{\triangle\}$
- $\delta_M(q, a) = \begin{cases} (i_A, R) & q = i_M \wedge a = \triangle \\ (\delta(q, a), R) & q \in Q_A \wedge a \in \Sigma_A \\ (f_M, \triangle) & q \in F_A \wedge a = \triangle \end{cases}$
- $i_M \in Q_M$ je počáteční stav stroje
- $f_M \in Q_M$ je koncový stav stroje

Ze struktury přechodové funkce Turingova stroje M lze vidět, že pro $w \in L$ o délce $n \in \mathbb{N}$, které je zapsáno na pásce, musí stroj po $1+n+1$ krocích přijmout v koncovém stavu f_M, nebo zhavarovat v kroku $m \leq n+2$. Turingův stroj tedy provede maximálně $n+2$ kroků, z čehož vyplývá, že jeho časová složitost je $O_{time}(n+2) = O_{time}(n)$. Protože při kontrole slova w musí stroj M projít právě buňky pásky, na kterých se slovo nachází, je jeho prostorová složitost $O_{space}(n+2) = O_{time}(n)$.

Bylo tedy dokázáno, že pro libovolný jazyk $L \in \mathcal{L}_3$ platí $L \in DTIME(n)$ a $L \in DSPACE(n)$. □