

Algoritmy a datové struktury I

2. cvičení

čtvrtek 3. 3. 2016 9:00

Fibonacciho čísla

Navrhněte algoritmus, který co nejrychleji spočte n -té Fibonacciho číslo.

Výpočetní model – RAM

Random Access Machine (RAM):

- paměť je rozdělena na buňky, které obsahují celá čísla a jsou celými čísly indexovány
- instrukce:
 - aritmetické a logické operace (jako v C) – $X := Y \text{ op } Z$
 - nepodmíněné skoky – `goto LABEL` (náveští se definuje napsáním `LABEL:` před instrukcí)
 - podmíněné (dle relačních operátorů) – `if X relop Y then instrukce`
 - zastavení výpočtu – `halt`
- operandy instrukcí jsou:
 - konstanty zapsané přímo v instrukci (např. 5)
 - přímo adresované buňky paměti (např. [5])
 - nepřímo adresované buňky paměti (např. [[5]])
 - pro pohodlí používáme pro buňky [-1] až [-26] zkratky A až Z
- cena instrukcí:
 - jednotková – příliš silný model
 - logaritická v hodnotě operandů
 - jednotková s (polynomiálně) omezenými čísly – nejběžnější

Asymptotická notace

- $\mathcal{O}(g) := \{f : (\exists c > 0)(\exists n_0 \in \mathbb{N})(\forall n > n_0)(|f(n)| \leq cg(n))\}$
- $\Omega(g) := \{f : (\exists c > 0)(\exists n_0 \in \mathbb{N})(\forall n > n_0)(|f(n)| \geq cg(n))\}$
- $\Theta(g) := \{f : (\exists c_0, c_1 > 0)(\exists n_0 \in \mathbb{N})(\forall n > n_0)(c_0g(n) \leq |f(n)| \leq c_1g(n))\}$
- $o(g) := \{f : (\forall c > 0)(\exists n_0 \in \mathbb{N})(\forall n > n_0)(|f(n)| \leq cg(n))\}$
- $\omega(g) := \{f : (\forall c > 0)(\exists n_0 \in \mathbb{N})(\forall n > n_0)(|f(n)| \geq cg(n))\}$

Ač je formálně lepší psát $f \in \mathcal{O}(g)$ (protože $\mathcal{O}(g)$ je třída funkcí), tak se píše $f = \mathcal{O}(g)$.

Příklady

1. Naprogramujte následující algoritmus na RAMu a zanalyzujte jeho časovou složitost (nejprve přesně, poté asymptoticky).
 - (1) while $N > 0$:
 - (2) if $N \bmod 2 = 0$: $N \leftarrow N \operatorname{div} 2$
 - (3) else: $N \leftarrow N - 1$
2. Jak nejrychleji umíte otestovat, zda číslo je prvočíslo.
3. Jak nejrychleji umíte najít všechna prvočísla $\leq n$.

Domácí úkoly

Úkoly jsou za plný počet bodů 14 dnů od zadání (deadline je počátek cvičení), poté za polovinu bodů. Úkoly mi pošlete na `husek+ads@iuuk.mff.cuni.cz`.

1. Jak spočítat celočíselnou druhou odmocninu? Tím pro dané x myslíme přiřazené číslo y takové, že $y^2 \leq x < (y + 1)^2$. [sqrt, 7]
2. Naprogramujte na RAMu nějaký třídící algoritmus se složitostí $\mathcal{O}(N \log N)$ nebo Quick-sort. Nezapomeňte popsat uložení dat v paměti. [ram, 10]
3. Máte posloupnosti $(x_i)_{i=0}^n$ a $(y_j)_{j=0}^m$ a číslo k . Vaším úkolem je najít i a j takové, že $x_i + y_j = k$ (pokud existují). [sum, 8]