1. CVIČENÍ Z ADS1

cvičí Martin Böhm, bohm@atrey.karlin.mff.cuni.cz

Příklad první

Vymyslete (nebo vzpomeňte si na) funkci, která roste asymptoticky rychleji než kterákoli jiná funkce zapsaná pomocí funkčního parametru, sčítání (odčítání), násobení, dělení a mocnění. Vymyslete také algoritmus, který ji počítá.

Příklad druhý

Rozhodněte o pravdivosti následujících výroků:

- $n^2 \in \Omega(n)$.
- $3n^2 + 5n + o(1) \in O(n^2)$.
- $\log n \in O(n^{0.001})$.
- $\log n + \Theta(n) \in \Theta(n)$.
- $n! \in O(2^n)$.
- $2^{2n} \in O(2^n)$.
- $2^{2^n} \in O((2^n)^2)$.

Příklad třetí

Rozmyslete si příklad nějakého (nearitmetického) algoritmu nebo datové struktury, která lze provést s nějakou složitostí na výpočetním modelu RAM, ale nelze provést na PM.

Příklad čtvrtý

Zkuste vymyslet, jak podvádět ve výpočetním modelu, kde je aritmetika libovolných čísel konstantně rychlá. Vymyslete nějaký zajímavý (a jiný, než aritmetický) algoritmus, který je asymptotický rychlejší, než co zvládneme na RAMu.

Můžete zkusit třeba nějaké jednoduché operace s polem délky n.

Příklad pátý

Klasické existenční otázky. Jaká je mohutnost množiny všech funkcí z přirozených čísel opět do přirozených čísel? Jaká je mohutnost množiny všech programů, které jdou naprogramovat v libovolném programovacím jazyce?

Odvoďte, že existuje funkce, která nelze v žádném programovacím jazyce naprogramovat.

Příklad šestý

Máme binární číslo n, chceme pomocí postupného přičtení k jedniček dostat číslo $k + n$, avšak nejsme na RAMu a každá operace změny cifry z jedničky na nulu nebo z nuly na jedničku nás stojí jednu operaci. Přičítání jedniček děláme klasicky, „školně“.

Jakou bude mít tento algoritmus složitost?

Příklad sedmý

Uvažujme klasický Eukleidův algoritmus s modulením. Nalezněte pro každé k lexikograficky nejmenší dvojici $(a, b), a < b$ takovou, že Eukleidův algoritmus pro spočítání $NSD(a, b)$ potřebuje právě k rekurzivních kroků.

Pomocí této množiny dvojic (znovu)-odvoďte složitost Eukleidova algoritmu.