

1. *Rychlý Fibonacci*: Jak rychle umíte spočítat n -té Fibonacciho číslo?
2. *Násobnost*: Na vstupu jsme dostali uspořádanou posloupnost čísel $x_1 \leq x_2 \leq \dots \leq x_n$. Jak upravit binární vyhledávání tak, aby vrátilo indexy prvního a posledního výskytu pro dané číslo k ?
3. *Děravá posloupnost*: Na vstupu jsme dostali rostoucí posloupnost přirozených čísel. Chceme najít nejmenší přirozené číslo, které v ní chybí. A co když budeme chtít najít k -té chybějící číslo?
4. *Hledání bez hranic*: Mějme neklesající funkci $f : \mathbb{N} \rightarrow \mathbb{N}$. Popište algoritmus, který najde nejmenší přirozené m takové, že $f(m) = k$. Nemáme žádný horní odhad na m a chceme časovou složitost vyjádřit v závislosti na m .
5. *Rychlý odhad*: Uvažujeme stejnou situaci jako v předchozím případě. Nyní nepožadujeme přesnou hodnotu m ale vystačíme si s odhadem. Nechť odhad m je přirozené k takové, že $2^{k-1} \leq m < 2^k$. Umíte najít odhad m rychleji než přesnou hodnotu?
6. *Odmocnina*: Spočítejte celočíselnou odmocninu z čísla x . To je největší y takové, že $y^2 \leq x$.