

Obecná notace: $\mathbb{N} = \{0, 1, 2, \dots\}$, $\log x = \log_2 x$

Asymptotická notace: Pro $f: \mathbb{N} \rightarrow \mathbb{R}_0^+$ def.

$$\mathcal{O}(f) = \{g: \mathbb{N} \rightarrow \mathbb{N} \mid \exists c \in \mathbb{R}^+, n_0 \in \mathbb{N}: \forall n \geq n_0: g(n) \leq cf(n)\}$$

$$\Omega(f) = \{g: \mathbb{N} \rightarrow \mathbb{N} \mid \exists c \in \mathbb{R}^+, n_0 \in \mathbb{N}: \forall n \geq n_0: g(n) \geq cf(n)\}$$

$$\Theta(f) = \mathcal{O}(f) \cap \Omega(f)$$

1. Seřadte výrazy podle inkluze tříd $\mathcal{O}(\dots)$, určete rovnosti:

$\bullet n^{2n}$	$\bullet n!$	$\bullet \log n$	$\bullet n^2$	$\bullet 2n$
$\bullet 2^n$	$\bullet n^{n+1}$	$\bullet \log_\pi n$	$\bullet 2^{n+\log n}$	$\bullet n^n$
$\bullet \sqrt{n}$	$\bullet n$	$\bullet 2^{n+1}$	$\bullet \log n^2$	$\bullet \log^2 n$

2. Dokažte „tranzitivitu“ $\mathcal{O}$, tedy $f \in \mathcal{O}(g) \wedge g \in \mathcal{O}(h) \Rightarrow f \in \mathcal{O}(h)$.

3. Dokažte, že pro $f, g: \mathbb{N} \rightarrow \mathbb{R}_0^+$ platí $\mathcal{O}(f + g) = \mathcal{O}(\max(f, g))$
(kde $\max(f, g)(x) = \max(f(x), g(x))$).

4. Jsou všechny funkce $f: \mathbb{N} \rightarrow \mathbb{R}_0^+$ porovnatelné inkluzí tříd $\mathcal{O}(f)$?
(Jinak řečeno, platí $\forall f, g: f \in \mathcal{O}(g) \vee g \in \mathcal{O}(f)$?)

5. Popište algoritmus, který pro nezakořeněný strom na vstupu najde kořen takový, že výsledný zakořeněný strom bude mít minimální výšku. Určete asymptotickou časovou složitost algoritmu.

6. Uvažme verzi QuickSortu, která jako pivot vezme vždy hodnotu prostředního prvku. Najděte pro každé n vstup délky n , který potrvá setřídít $\mathcal{O}(n^2)$ času.