Kapitola 1

Úvod

1.1 Základní pojmy

1.1.1 Algoritmus. *Algoritmem* rozumíme dobře definovaný proces, tj. posloupnost výpočetních kroků, který přijímá hodnoty (zadání, vstup) a vytváří hodnoty (řešení, výstup).

1.1.2 Problém, úloha. *Úloha*, též *problém*, je obecná specifikace vztahu zadání/řešení. *Instancí* problému, úlohy $\mathcal{U}$ rozumíme konkrétní zadání všech parametrů, které daná úloha (problém) obsahuje. Jinými slovy, instance úlohy je správný příklad zadání.

1.1.3 Řekneme, že algoritmus $\mathcal{A}$ *řeší* úlohu $\mathcal{U}$, jestliže pro každý vstup (každou instanci problému $\mathcal{U}$) vydá správné řešení.

Poznamenejme, že předchozí věta znamená, že každý algoritmus, který řeší nějakou úlohu, se vždy zastaví. To znamená, že algoritmus, který se na nějakém vstupu nezastaví, nemůže řešit žádnou úlohu.

1.1.4 Analýza časové složitosti algoritmu. Existují dva základní způsoby měření časové náročnosti algoritmů.

1. Analýza nejhoršího případu. Jedná se o asymptotický odhad $T(n)$ času potřebného pro vyřešení každé instance velikosti n.

2. Průměrná složitost. Jedná se o asymptotický odhad $T_{aver}(n)$ průměrného času, který je potřeba pro vyřešení instance velikosti n, kde bereme v úvahu s jakou pravděpodobností se jednotlivé instance (typy instancí) vyskytují.

Pro posloupnost operací stejného druhu používáme ještě amortizovanou složitost. Amortizovaná složitost je průměrná složitost nejhoršího případu pro posloupnost n operací/instrukcí stejného druhu.

1.2 Asymptotický růst funkcí

Připomeňme základní pojmy týkající se růstu nezáporných funkcí.

1.2.1 Symbol $\mathcal{O}$. Je dána nezáporná funkce $g(n)$. Řekneme, že nezáporná funkce $f(n)$ *je* $\mathcal{O}(g(n))$, jestliže existuje kladná konstanta c a přirozené číslo n_0 tak, že

$$f(n) \le c\, g(n) \quad \text{pro všechny } n \ge n_0.$$

□

$\mathcal{O}(g(n))$ můžeme též chápat jako třídu všech nezáporných funkcí $f(n)$:

$$\mathcal{O}(g(n)) = \{f(n) \mid \exists c > 0, n_0 \in \mathbb{N} \text{ tak, že } f(n) \le c\, g(n) \;\; \forall n \ge n_0\}.$$

1.2.2 Symbol Ω. Je dána nezáporná funkce $g(n)$. Řekneme, že nezáporná funkce $f(n)$ *je* $\Omega(g(n))$, jestliže existuje kladná konstanta c a přirozené číslo n_0 tak, že

$$f(n) \ge c\, g(n) \quad \text{pro všechny } n \ge n_0.$$

□

$\Omega(g(n))$ můžeme též chápat jako třídu všech nezáporných funkcí $f(n)$:

$$\Omega(g(n)) = \{f(n) \mid \exists c > 0, n_0 \in \mathbb{N} \text{ tak, že } f(n) \ge c\, g(n) \;\; \forall\, n \ge n_0\}.$$

1.2.3 Poznámka. Fakt, že funkce $f(n)$ je $\Omega(g(n))$ je ekvivalentní faktu, že funkce $g(n)$ je $\mathcal{O}(f(n))$.

1.2.4 Symbol Θ. Je dána nezáporná funkce $g(n)$. Řekneme, že nezáporná funkce $f(n)$ *je* $\Theta(g(n))$, jestliže existují kladné konstanty c_1, c_2 a přirozené číslo n_0 tak, že

$$c_1\, g(n) \le f(n) \le c_2\, g(n) \quad \text{pro všechny } n \ge n_0.$$

□

$\Theta(g(n))$ můžeme též chápat jako třídu všech nezáporných funkcí $f(n)$:

$$\Theta(g(n)) = \{f(n) \mid \exists c_1, c_2 > 0, n_0 \in \mathbb{N} \text{ tak, že } c_1\, g(n) \le f(n) \le c_2\, g(n) \;\; \forall\, n \ge n_0\}.$$

1.2.5 Poznámka. Platí $f(n)$ je $\Theta(g(n))$ právě tehdy, když $f(n)$ je zároveň $\mathcal{O}(g(n))$ a $\Omega(g(n))$.

V dalším zavedeme ještě dvě další třídy funkcí, totiž $o(g(n))$ a $\omega(g(n))$.

1.2.6 Symbol malé o. Je dána nezáporná funkce $g(n)$. Řekneme, že nezáporná funkce $f(n)$ *je* $o(g(n))$, jestliže pro každou kladnou konstantu c existuje přirozené číslo n_0 tak, že

$$0 \le f(n) < c\, g(n) \quad \text{pro všechny } n \ge n_0.$$

□

$o(g(n))$ můžeme též chápat jako třídu všech nezáporných funkcí $f(n)$:

$$o(g(n)) = \{f(n) \mid \forall\, c > 0\; \exists\, n_0 \in \mathbb{N} \text{ tak, že } 0 \le f(n) < c\, g(n) \;\; \forall\, n > n_0\}.$$

1.2.7 Poznámka. Fakt, že nezáporná funkce $f(n)$ je $\mathcal{O}(g(n))$, zhruba řečeno znamená, že funkce $f(n)$ neroste asymptoticky více než funkce $g(n)$. Naproti tomu fakt, že nezáporná funkce $f(n)$ je $o(g(n))$, znamená, že funkce $f(n)$ roste asymptoticky méně než funkce $g(n)$.

1.2.8 Symbol malé ω. Je dána nezáporná funkce $g(n)$. Řekneme, že nezáporná funkce $f(n)$ *je* $\omega(g(n))$, jestliže pro každou kladnou konstantu c existuje přirozené číslo n_0 tak, že

$$0 \leq c\,g(n) < f(n) \quad \text{pro všechny } n \geq n_0.$$

□

$\omega(g(n))$ můžeme též chápat jako třídu všech nezáporných funkcí $f(n)$:

$$\omega(g(n)) = \{f(n) \mid \forall\, c > 0\, \exists\, n_0 \in \mathbb{N} \text{ tak, že } 0 \leq c\,g(n) < f(n) \;\; \forall\, n > n_0\}.$$

1.2.9 Poznámka. Fakt, že nezáporná funkce $f(n)$ je $\Omega(g(n))$, zhruba řečeno znamená, že funkce $f(n)$ roste asymptoticky alespoň tak, jako funkce $g(n)$. Naproti tomu fakt, že nezáporná funkce $f(n)$ je $\omega(g(n))$, znamená, že funkce $f(n)$ roste asymptoticky více než funkce $g(n)$.

1.2.10 Značení. Protože symboly $\mathcal{O}, \Omega, \Theta$ představují množiny funkcí, budeme v dalším textu psát $f(n) \in \mathcal{O}(g(n))$. Je ovšem pravda, že v literatuře najdete i zápis $f(n) = \mathcal{O}(g(n))$. Při tomto zápisu je třeba mít na paměti, že znak rovnosti v zápise $f(n) = \mathcal{O}(g(n))$ nemá stejné vlastnosti jako klasická rovnost. Obdobně pro ostatní symboly.

1.2.11 Tvrzení. Jsou dány dvě nezáporné funkce $f(n)$ a $g(n)$. Pak platí

1. $f(n) \in o(g(n))$ právě tehdy, když $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$;

2. $f(n) \in \omega(g(n))$ právě tehdy, když $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$.

3. Jestliže $\lim_{n\to\infty} \frac{f(n)}{g(n)} = a$ pro některé $a \in \mathbb{R}$, $a \neq 0$, pak $f(n) \in \Theta(g(n))$

Zdůvodnění: 1) Napíšeme, co znamená fakt $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$:

$$\forall\, \varepsilon > 0\, \exists\, n_0 \in \mathbb{N} \;\text{ tak, že }\; \forall\, n \geq n_0 \;\text{ platí }\; \left|\frac{f(n)}{g(n)}\right| < \varepsilon.$$

Vztah $|\frac{f(n)}{g(n)}| < \varepsilon$ lze přepsat na $f(n) < \varepsilon\, g(n)$. Označíme-li $c := \varepsilon$, dostáváme $f(n)$ je $o(g(n))$.

3) Obdobně fakt $\lim_{n\to\infty} \frac{f(n)}{g(n)} = a$, $a > 0$, znamená, že

$$\forall\, \varepsilon > 0\, \exists\, n_0 \in \mathbb{N} \;\text{ tak, že }\; \forall\, n \geq n_0 \;\text{ platí }\; \left|\frac{f(n)}{g(n)} - a\right| < \varepsilon.$$

Jinak zapsáno $(a - \varepsilon)g(n) < f(n) < (a + \varepsilon)g(n)$. Zvolíme-li $\varepsilon = \frac{a}{2}$, dostáváme

$$\frac{a}{2}\, g(n) < f(n) < \frac{3a}{2}\, g(n);$$

tedy $f(n)$ je $\Theta(g(n))$. □

1.2.12 Tranzitivita. Není těžké se přesvědčit, že platí následující tvrzení.

Tvrzení. Máme dány tři nezáporné funkce $f(n)$, $g(n)$ a $h(n)$.

1. Jestliže $f(n) \in \mathcal{O}(g(n))$ a $g(n) \in \mathcal{O}(h(n))$, pak $f(n) \in \mathcal{O}(h(n))$.
2. Jestliže $f(n) \in \Omega(g(n))$ a $g(n) \in \Omega(h(n))$, pak $f(n) \in \Omega(h(n))$.
3. Jestliže $f(n) \in \Theta(g(n))$ a $g(n) \in \Theta(h(n))$, pak $f(n) \in \Theta(h(n))$.

1.2.13 Reflexivita. Pro všechny nezáporné funkce $f(n)$ platí: $f(n) \in \mathcal{O}(f(n))$, $f(n) \in \Omega(f(n))$ a $f(n) \in \Theta(f(n))$.

1.2.14 Tvrzení. $f(n) \in \Theta(g(n))$ právě tehdy, když $g(n) \in \Theta(f(n))$. □

1.2.15 Příklady.

1. Pro každé $a > 1$ a $b > 1$ platí
$$\log_a(n) \in \Theta(\log_b(n)).$$
2. V celém textu značíme logaritmus o základu 2 symbolem lg, tj. $\lg(n) = \log_2(n)$. Platí
$$\lg n! \in \Theta(n \lg n).$$

Druhá část tvrzení vyplývá např. z následující věty.

1.2.16 Věta (Gauss). Pro každé $n \geq 1$ platí
$$n^{\frac{n}{2}} \leq n! \leq \left(\frac{n+1}{2}\right)^n.$$
□

Zdůvodnění: Využijeme fakt, že pro každá dvě kladná čísla a, b platí $\frac{a+b}{2} \geq \sqrt{ab}$.

Přepíšeme $(n!)^2$ takto
$$(n!)^2 = n\,(n-1)\,\ldots 2 \cdot 1 \cdot 1 \cdot 2 \ldots\,(n-1)\,n = \prod_{i=1}^{n}(n-i+1)i.$$

Odtud
$$n! = \prod_{i=1}^{n}\sqrt{(n-i+1)i} \leq \prod_{i=1}^{n}\frac{n+1}{2} = \left(\frac{n+1}{2}\right)^n,$$
protože pro každé i platí $\sqrt{(n-i+1)i} \leq \frac{n-i+1+i}{2}$. Tím jsme dostali horní odhad.

Na druhé straně pro každé i platí $n \leq (n-i+1)i$ a proto je $n^n \leq (n!)^2$. Odmocněním dostaneme dolní odhad, totiž $n^{\frac{n}{2}} \leq n!$.

1.2.17 Věta. Máme dánu nezápornou funkci $f(n)$, která je neklesající. Jestliže platí $f(\frac{n}{2}) \in \Theta(f(n))$, pak

$$\sum_{i=1}^{n} f(i) \in \Theta(n\, f(n)).$$

□

Krátké zdůvodnění: Fakt, že $\sum_{i=1}^{n} f(i) \in \mathcal{O}(n\, f(n))$ je zřejmý: f je neklesající.

Dále existuje kladná konstanta c taková, že pro dostatečně velká n platí $c\, f(n) \leq f(\frac{n}{2})$. Proto platí

$$\sum_{i=1}^{n} f(i) \geq f(\frac{n}{2}) + \ldots + f(n) \geq \frac{n}{2}\, c\, f(n).$$

To znamená, že $\sum_{i=1}^{n} f(i) \geq \frac{c}{2}\, n\, f(n)$ a proto $\sum_{i-1}^{n} f(i) \in \Omega(n\, f(n))$.

1.2.18 Poznámka. Vlastnost z předchozí věty má např. funkce $f(n) = n^d$ pro přirozené číslo $d \geq 1$, nemá ji však funkce $f(n) = 2^n$. Pro asymptotický odhad $\sum_{i=1}^{n} 2^i$ se dá využít následující metoda:

Matematickou indukcí dokážeme, že existuje $c > 0$ takové, že

$$\sum_{i=1}^{n} 2^i \leq c \cdot 2^n.$$

Základní krok. Víme, že $\sum_{i=1}^{1} 2^i = 2$ a $2 \leq c \cdot 2$ pro každou konstantu $c \geq 1$.
Indukční krok. Předpokládejme, že platí $\sum_{i=1}^{n} 2^i \leq c \cdot 2^n$. Pak

$$\sum_{i=1}^{n+1} 2^i = \sum_{i=1}^{n} 2^i + 2^{n+1} \leq c\, 2^n + 2^{n+1} = \left(\frac{1}{2} + \frac{1}{c}\right) c \cdot 2^{n+1}.$$

Nyní k dokončení důkazu stačí zajistit, aby $\frac{1}{2} + \frac{1}{c} \leq 1$. A to je ekvivalentní s podmínkou $c \geq 2$.

1.2.19 Ještě jeden způsob získání odhadu. Tvrzení. Pro neklesající nezápornou funkci $f(n)$ platí

$$\int_0^n f(x)\, dx \leq \sum_{i=1}^{n} f(i) \leq \int_1^{n+1} f(x)\, dx.$$

Pro nerostoucí nezápornou funkci $f(n)$ platí

$$\int_1^{n+1} f(x)\, dx \leq \sum_{i=1}^{n} f(i) \leq \int_0^{n} f(x)\, dx.$$

K tomu, abychom se přesvědčili o výše uvedeném, stačí si nakreslit obrázek a součet $\sum_{i=1}^{n} f(i)$ si představit jako součet ploch obdélníků s jednou stranou 1 a jednou stranou $f(i)$. □

Poznamenejme, že v některých případech $\int_0^n f(x)\, dx$ může být nevlastní integrál. To řešíme tak, že použijeme vztah $\sum_{i=1}^{n} f(i) = f(1) + \sum_{i=2}^{n} f(i)$ a teprve $\sum_{i=2}^{n} f(i)$ omezujeme integrálem.