Obsah

Fibonacciho posloupnost

Fibonacciho posloupnost

$$F(n) = \begin{cases} 0, & \text{pro n} = 0 \\ 1, & \text{pro n} = 1 \\ F(n-1) + F(n-2), & \text{pro n} \geq 2 \end{cases}$$

Fibonacciho posloupnost

$$F(n) = \begin{cases} 0, & \text{pro n} = 0 \\ 1, & \text{pro n} = 1 \\ F(n-1) + F(n-2), & \text{pro n} \geq 2 \end{cases}$$

0	0
1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34

Fibonacciho posloupnost

$$F(n) = \begin{cases} 0, & \text{pro n} = 0 \\ 1, & \text{pro n} = 1 \\ F(n-1) + F(n-2), & \text{pro n} \geq 2 \end{cases}$$

0	0
1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34

10	55
11	89
12	144
13	233
14	377
15	610
16	987
17	1597
18	2584
19	4181

Fibonacciho posloupnost

$$F(n) = \begin{cases} 0, & \text{pro n} = 0 \\ 1, & \text{pro n} = 1 \\ F(n-1) + F(n-2), & \text{pro n} \geq 2 \end{cases}$$

0	0
1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34

10	55
11	89
12	144
13	233
14	377
15	610
16	987
17	1597
18	2584
19	4181

20	6765
30	832040
40	102334155
50	12586269025
60	1548008755920
70	190392490709135
80	23416728348467685
90	2880067194370816120
100	354224848179261915075
110	43566776258854844738105

Fibonacciho posloupnost

Rekurzivní řešení (naivní)

Fibonacciho posloupnost

Rekurzivní řešení (naivní)

$\mathcal{O}(2^n)$

Fibonacciho posloupnost

Rekurzivní řešení (naivní)

$\mathcal{O}(2^n)$

Iterativní řešení

Fibonacciho posloupnost

Rekurzivní řešení (naivní)

$\mathcal{O}(2^n)$

Iterativní řešení

$\mathcal{O}(n)$

Fibonacciho posloupnost

Rekurzivní řešení (naivní)

$\mathcal{O}(2^n)$

Iterativní řešení

$\mathcal{O}(n)$

Explicitní vyjádření

Fibonacciho posloupnost

Rekurzivní řešení (naivní)

$\mathcal{O}(2^n)$

Iterativní řešení

$\mathcal{O}(n)$

Explicitní vyjádření

$\mathcal{O}(1)$

Asymptotická složitost

Asymptotická složitost je způsob **klasifikace počítačových algoritmů**. Určuje operační náročnost algoritmu tak, že zjišťuje, jakým způsobem se bude chování algoritmu měnit **v závislosti na změně velikosti** (počtu) **vstupních dat**. [Wikipedia]

Asymptotická složitost

	n	$n \log_2 n$	n^2	n^3	1.5^n	2^n	$n!$
$n = 10$	< 1 sec	< 1 sec	< 1 sec	< 1 sec	< 1 sec	< 1 sec	4 sec
$n = 30$	< 1 sec	< 1 sec	< 1 sec	< 1 sec	< 1 sec	18 min	10^{25} years
$n = 50$	< 1 sec	< 1 sec	< 1 sec	< 1 sec	11 min	36 years	very long
$n = 100$	< 1 sec	< 1 sec	< 1 sec	1 sec	12,892 years	10^{17} years	very long
$n = 1{,}000$	< 1 sec	< 1 sec	1 sec	18 min	very long	very long	very long
$n = 10{,}000$	< 1 sec	< 1 sec	2 min	12 days	very long	very long	very long
$n = 100{,}000$	< 1 sec	2 sec	3 hours	32 years	very long	very long	very long
$n = 1{,}000{,}000$	1 sec	20 sec	12 days	31,710 years	very long	very long	very long

[http://modelingsocialdata.org/lectures/2017/02/03/lecture-3-computational-complexity.html]

Příklad výpočetní náročnosti [Wikipedia]

Probírané asymptotické notace

$$\mathcal{O}, \Omega, \Theta$$

Asymptotická horní mez / $\mathcal{O}$ / big O

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

Asymptotická horní mez / $\mathcal{O}$ / big O

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n \overset{?}{\in} \mathcal{O}(n^2)$$

Asymptotická horní mez / $\mathcal{O}$ / big O

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n \stackrel{?}{\in} \mathcal{O}(n^2)$$

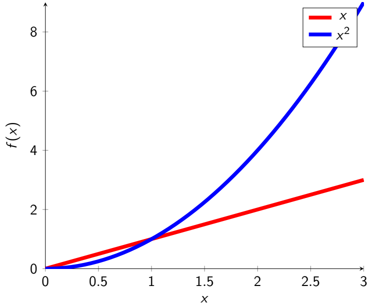
Ano, například $c = 1, n_0 = 1$

Asymptotická horní mez / $\mathcal{O}$ / big O

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n \overset{?}{\in} \mathcal{O}(n^2)$$

Ano, například $c = 1, n_0 = 1$



Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n \overset{?}{\in} \mathcal{O}(0.5n)$$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n \overset{?}{\in} \mathcal{O}(0.5n)$$

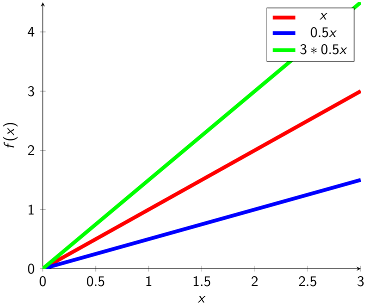
Ano, například $c = 3, n_0 = 1$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n \overset{?}{\in} \mathcal{O}(0.5n)$$

Ano, například $c = 3, n_0 = 1$



Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n^2 + 1 \overset{?}{\in} \mathcal{O}(n^2)$$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n^2 + 1 \overset{?}{\in} \mathcal{O}(n^2)$$

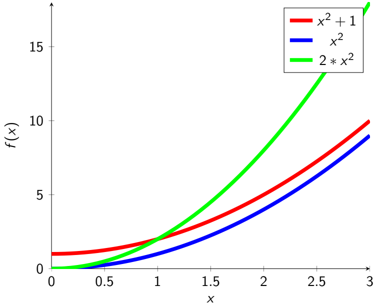
Ano, například $c = 2, n_0 = 1$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$n^2 + 1 \stackrel{?}{\in} \mathcal{O}(n^2)$$

Ano, například $c = 2, n_0 = 1$



Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$2^{n+5} \stackrel{?}{\in} \mathcal{O}(2^n)$$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$2^{n+5} \stackrel{?}{\in} \mathcal{O}(2^n)$$

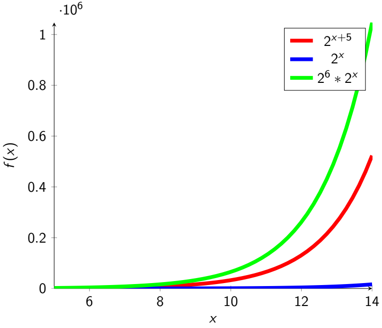
Ano, například $c = 2^6, n_0 = 10$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$2^{n+5} \overset{?}{\in} \mathcal{O}(2^n)$$

Ano, například $c = 2^6$, $n_0 = 10$



Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$1.5n + 2.2n^2 + 0.001n^3 \stackrel{?}{\in} \mathcal{O}(n^3)$$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$1.5n + 2.2n^2 + 0.001n^3 \overset{?}{\in} \mathcal{O}(n^3)$$

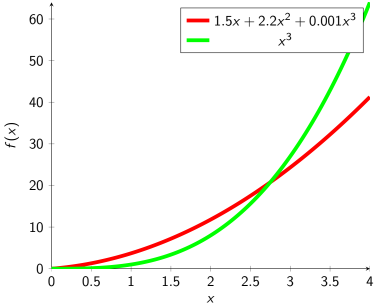
Ano, například $c = 1, n_0 = 3$

Asymptotická horní mez / $\mathcal{O}$ příklady

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$1.5n + 2.2n^2 + 0.001n^3 \overset{?}{\in} \mathcal{O}(n^3)$$

Ano, například $c = 1, n_0 = 3$



Asymptotická horní mez / $\mathcal{O}$ neformální poučka

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj
- Ignoruj multiplikativní konstanty u nejrychleji rostoucího členu

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj
- Ignoruj multiplikativní konstanty u nejrychleji rostoucího členu
- Porovnej výsledky

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj
- Ignoruj multiplikativní konstanty u nejrychleji rostoucího členu
- Porovnej výsledky

$$1.5n + 2.2n^2 + 0.001n^3 \stackrel{?}{\in} \mathcal{O}(n^3)$$

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj
- Ignoruj multiplikativní konstanty u nejrychleji rostoucího členu
- Porovnej výsledky

$$1.5n + 2.2n^2 + 0.001n^3 \overset{?}{\in} \mathcal{O}(n^3)$$

V

$$1.5n + 2.2n^2 + 0.001n^3$$

je nejryhleji rostoucím členem $0.001n^3$

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj
- Ignoruj multiplikativní konstanty u nejrychleji rostoucího členu
- Porovnej výsledky

$$1.5n + 2.2n^2 + 0.001n^3 \overset{?}{\in} \mathcal{O}(n^3)$$

V

$$1.5n + 2.2n^2 + 0.001n^3$$

je nejryhleji rostoucím členem $0.001n^3$
Bez multiplikativní konstanty je to pouze n^3

Asymptotická horní mez / $\mathcal{O}$ neformální poučka

- V polynomech najdi nejrychleji rostoucí člen
- Ostatní členy ignoruj
- Ignoruj multiplikativní konstanty u nejrychleji rostoucího členu
- Porovnej výsledky

$$1.5n + 2.2n^2 + 0.001n^3 \overset{?}{\in} \mathcal{O}(n^3)$$

V

$$1.5n + 2.2n^2 + 0.001n^3$$

je nejryhleji rostoucím členem $0.001n^3$
Bez multiplikativní konstanty je to pouze n^3
Tudíž n^3 je asymptotickou horní mezí $1.5n + 2.2n^2 + 0.001n^3$

Asymptotická spodní mez / Ω

Téměř opak $\mathcal{O}$

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

Asymptotická spodní mez / Ω příklady

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^{+})(\exists n_0 \in \mathbb{N}^{+})(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

Asymptotická spodní mez / Ω příklady

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

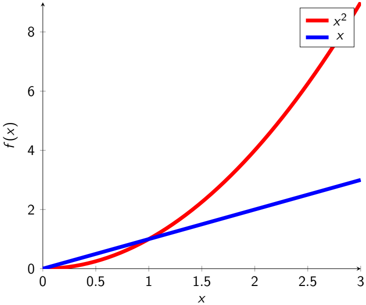
$$n^2 \overset{?}{\in} \Omega(n)$$

Asymptotická spodní mez / Ω příklady

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

$$n^2 \stackrel{?}{\in} \Omega(n)$$

Ano, například $c = 1, n_0 = 1$



Asymptotická spodní mez / Ω příklady

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

$$n^3 \overset{?}{\in} \Omega(1.5n + 2.2n^2 + 0.001n^3)$$

Asymptotická spodní mez / Ω příklady

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

$$n^3 \stackrel{?}{\in} \Omega(1.5n + 2.2n^2 + 0.001n^3)$$

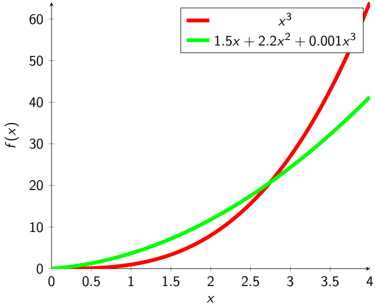
Ano, například $c = 1, n_0 = 3$

Asymptotická spodní mez / Ω příklady

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

$$n^3 \overset{?}{\in} \Omega(1.5n + 2.2n^2 + 0.001n^3)$$

Ano, například $c = 1, n_0 = 3$



Asymptotická těsná mez / Θ

Předchozí dvě dohromady, tedy musí platit $\mathcal{O}$ a zároveň i Ω

$$f(n) \in \mathcal{O}(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : f(n) \leq c \cdot g(n)$$

$$f(n) \in \Omega(g(n)) \iff (\exists c \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : c \cdot f(n) \leq g(n)$$

$$\begin{aligned} f(n) \in \Theta(g(n)) \iff & \qquad (1) \\ & (\exists c_1, c_2 \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : \qquad (2) \\ & c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \qquad (3) \end{aligned}$$

Asymptotická těsná mez / Θ příklad

$$f(n) \in \Theta(g(n)) \iff \tag{4}$$

$$(\exists c_1, c_2 \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0): \tag{5}$$

$$c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \tag{6}$$

$$\ln(n/2) \stackrel{?}{\in} \Theta(\ln(n))$$

Asymptotická těsná mez / Θ příklad

$$f(n) \in \Theta(g(n)) \iff \tag{4}$$
$$(\exists c_1, c_2 \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0) : \tag{5}$$
$$c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \tag{6}$$

$$\ln(n/2) \stackrel{?}{\in} \Theta(\ln(n))$$

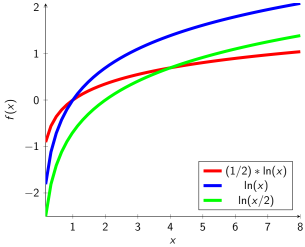
Ano, například $c_1 = 1/2, c_2 = 1, n_0 = 4$

Asymptotická těsná mez / Θ příklad

$$f(n) \in \Theta(g(n)) \iff \quad (4)$$
$$(\exists c_1, c_2 \in \mathbb{R}^+)(\exists n_0 \in \mathbb{N}^+)(\forall n \geq n_0): \quad (5)$$
$$c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n) \quad (6)$$

$$\ln(n/2) \stackrel{?}{\in} \Theta(\ln(n))$$

Ano, například $c_1 = 1/2, c_2 = 1, n_0 = 4$



Time complexity vs. Space complexity

Time complexity is the computational complexity that describes the amount of **computer time** it takes to run an algorithm.
Wikipedia

Space complexity of an algorithm or a computer program is the amount of **memory space** required to solve an instance of the computational problem.
Wikipedia

Insertion sort

- Náročnost
  - Nejhorší - $O(n^2)$ comparisons and swaps
  - Nejlepší - $O(n)$ comparisons, $O(1)$ swaps
- Jednoduchá implementace
- Efektivní na malé nebo předseřazené pole
- Online - lze prvky dodávat postupně
- Animace [Wikipedia]

Selection sort

- Náročnost
  - Nejhorší - $O(n^2)$ comparisons, $O(n)$ swaps
  - Nejlepší - $O(n^2)$ comparisons, $O(1)$ swaps
- Jednoduchá implementace
- Nestabilní řazení - prvky se stejnou hodnotou nemusí být ve vzájemně stejném pořadí po seřazení
- Seřazené pole se bude zpracovávat stejně dlouho jako neseřazené
- Animace [Wikipedia]

Bubble sort

- Náročnost
  - Nejhorší - $O(n^2)$ comparisons and swaps
  - Nejlepší - $O(n)$ comparisons, $O(1)$ swaps
- Jednoduchá implementace
- Používá se prakticky jen pro vyukové účely
- Zpracování probíhá "probubláváním" prvku s nejvyšší hodnotou na konec pole
- Animace [Wikipedia]

Quick sort

- Náročnost
  - Nejhorší - $O(n^2)$
  - Nejlepší - $O(n \log(n))$
- Složitejší implementace
- Nejrychlejší ze zmíněných pro velké pole
- Volba pivotu ovliňuje výkonost řazení
- Animace [Wikipedia]

Quick sort - volba pivotu

- První prvek
  - Velmi nevýhodné na částečně seřazených množinách
- Náhodný prvek
  - Nejčastěji používaná metoda
  - Nejhorší scénář je pořád $O(n^2)$
- Medián tří (případně více) prvků
  - Vyberou se náhodně z množiny a jako pivot se zvolí jejich medián

Garbage collector

- Automatická správa paměti
- Uvolňuje programem již nepoužívanou část paměti
- Součástí téměř všech moderních jazyků

Dynamické pole

```
// Pokud i == m
//      m = 2*m
//      Alokuj P' o velikosti m
//      Pro j, ..., i - 1:
//          P'[j] = P[j]
//      Dealokuj P
//      P = P'
// P[i] = x
// i = i + 1
```

- *Ačkoliv je složitost přidání jednoho prvku v nejhorším případě $\mathcal{O}(n)$, v posloupnosti operací se chová, jako kdyby byla konstantní. Budeme proto říkat, že je* **amortizovaně konstantní**.

Amortizovaná složitost

- Průměrná složitost algoritmu
- **Neposkytuje jistotu** jelikož musí být splněny předpoklady sekvence
  - (v konkrétním bodě amortizovaná složitost neplatí, v úseku již ano)

Amortizace

"Amortize" is a fancy verb used in finance that refers to paying off the cost of something gradually. With dynamic arrays, every expensive append where we have to grow the array "buys" us many cheap appends in the future. Conceptually, we can spread the cost of the expensive append over all those cheap appends.

Amortizovaná složitost vs. Asymptotická složitost

Asymptotická složitost

- Složitost $\mathcal{O}$ je určena na základě nejhorší možné instance běhu algoritmu

Amortizovaná složitost

- Amortizovaná časová složitost označuje časovou složitost algoritmu v **sekvenci** nejhorších možných vstupních dat
- Nevyužívá pravděpodobnosti ($\Rightarrow$ na sekvenci dat je zaručena)