Základy matematické analýzy

Složitost algoritmů

Jitka Hrabáková[1], Tomáš Kalvoda[2], Ivo Petr[3]

[1] jitka.hrabakova@fit.cvut.cz [2] tomas.kalvoda@fit.cvut.cz, [3] ivo.petr@fit.cvut.cz,

Katedra aplikované matematiky
Fakulta informačních technologií
České vysoké učení technické v Praze

4. září 2019
ZS 2018/2019

Hlavní body

1. Uspořádání
2. Složitost jednoduchých třídících algoritmů

Hlavní body

1 Uspořádání

2 Složitost jednoduchých třídících algoritmů

Uspořádání

Definice (relace):

Relací $\mathcal{R}$ na množine M nazýváme podmnožinu $M \times M$. Místo $(a, b) \in \mathcal{R}$ zkráceně píšeme $x\mathcal{R}y$.

Definice (uspořádání):

Relaci $\mathcal{R}$ na množině M splňující

Uspořádání

Definice (relace):

Relací $\mathcal{R}$ na množine M nazýváme podmnožinu $M \times M$. Místo $(a, b) \in \mathcal{R}$ zkráceně píšeme $x\mathcal{R}y$.

Definice (uspořádání):

Relaci $\mathcal{R}$ na množině M splňující

1. (reflexivita): pro každé $x \in M$ platí $x\mathcal{R}x$,

Uspořádání

Definice (relace):

Relací $\mathcal{R}$ na množine M nazýváme podmnožinu $M \times M$. Místo $(a, b) \in \mathcal{R}$ zkráceně píšeme $x\mathcal{R}y$.

Definice (uspořádání):

Relaci $\mathcal{R}$ na množině M splňující

1. (reflexivita): pro každé $x \in M$ platí $x\mathcal{R}x$,
2. (antisymetrie): pro každé $x, y \in M$ platí, že pokud $x\mathcal{R}y$ a $y\mathcal{R}x$ pak i $x = y$,

Uspořádání

Definice (relace):

Relací $\mathcal{R}$ na množine M nazýváme podmnožinu $M \times M$. Místo $(a, b) \in \mathcal{R}$ zkráceně píšeme $x\mathcal{R}y$.

Definice (uspořádání):

Relaci $\mathcal{R}$ na množině M splňující

1. (reflexivita): pro každé $x \in M$ platí $x\mathcal{R}x$,
2. (antisymetrie): pro každé $x, y \in M$ platí, že pokud $x\mathcal{R}y$ a $y\mathcal{R}x$ pak i $x = y$,
3. (tranzitivita): pokud pro $x, y, z \in M$ platí $x\mathcal{R}y$ a $y\mathcal{R}z$, pak platí $x\mathcal{R}z$,

Uspořádání

Definice (relace):

Relací $\mathcal{R}$ na množine M nazýváme podmnožinu $M \times M$. Místo $(a, b) \in \mathcal{R}$ zkráceně píšeme $x\mathcal{R}y$.

Definice (uspořádání):

Relaci $\mathcal{R}$ na množině M splňující

1. (reflexivita): pro každé $x \in M$ platí $x\mathcal{R}x$,
2. (antisymetrie): pro každé $x, y \in M$ platí, že pokud $x\mathcal{R}y$ a $y\mathcal{R}x$ pak i $x = y$,
3. (tranzitivita): pokud pro $x, y, z \in M$ platí $x\mathcal{R}y$ a $y\mathcal{R}z$, pak platí $x\mathcal{R}z$,

nazýváme **uspořádáním** na množině M.

Uspořádání

Definice (relace):

Relací $\mathcal{R}$ na množine M nazýváme podmnožinu $M \times M$. Místo $(a, b) \in \mathcal{R}$ zkráceně píšeme $x\mathcal{R}y$.

Definice (uspořádání):

Relaci $\mathcal{R}$ na množině M splňující

1. (reflexivita): pro každé $x \in M$ platí $x\mathcal{R}x$,
2. (antisymetrie): pro každé $x, y \in M$ platí, že pokud $x\mathcal{R}y$ a $y\mathcal{R}x$ pak i $x = y$,
3. (tranzitivita): pokud pro $x, y, z \in M$ platí $x\mathcal{R}y$ a $y\mathcal{R}z$, pak platí $x\mathcal{R}z$,

nazýváme **uspořádáním** na množině M.

Poznámka:

Relaci $\mathcal{R}$, jež je uspořádáním, většinou značíme symbolem $\leq$.

Příklady

Příklad.

Na množině $\mathbb{N}$ zaveďme relaci $\mathcal{R}$: $n\mathcal{R}m$ kdykoliv $\gcd(n,m) = 1$.
Jinak řečeno, n a m jsou spolu v relaci $\mathcal{R}$, právě když jsou nesoudělná.

- $(2,3) \in \mathcal{R}$, tj. $2\mathcal{R}3$ je pravdivé.
- $(2,4) \notin \mathcal{R}$, tj. $2\mathcal{R}4$ není pravdivé.
- Tato relace není relací uspořádání. Proč?

Příklady

Příklad.

Na množině $\mathbb{N}$ zaveďme relaci $\mathcal{R}$: $n\mathcal{R}m$ kdykoliv $\gcd(n,m) = 1$.
Jinak řečeno, n a m jsou spolu v relaci $\mathcal{R}$, právě když jsou nesoudělná.

- $(2,3) \in \mathcal{R}$, tj. $2\mathcal{R}3$ je pravdivé.
- $(2,4) \notin \mathcal{R}$, tj. $2\mathcal{R}4$ není pravdivé.
- Tato relace není relací uspořádání. Proč?

Příklad.

Je-li M množina reálných čísel a $x\mathcal{R}y$ znamená „x je menší nebo rovno y“, pak $\mathcal{R}$ představuje uspořádání na množině $\mathbb{R}$.

Příklady

Příklad.

Na množině $\mathbb{N}$ zaveďme relaci $\mathcal{R}$: $n\mathcal{R}m$ kdykoliv $\gcd(n, m) = 1$.
Jinak řečeno, n a m jsou spolu v relaci $\mathcal{R}$, právě když jsou nesoudělná.

- $(2, 3) \in \mathcal{R}$, tj. $2\mathcal{R}3$ je pravdivé.
- $(2, 4) \notin \mathcal{R}$, tj. $2\mathcal{R}4$ není pravdivé.
- Tato relace není relací uspořádání. Proč?

Příklad.

Je-li M množina reálných čísel a $x\mathcal{R}y$ znamená „x je menší nebo rovno y“, pak $\mathcal{R}$ představuje uspořádání na množině $\mathbb{R}$.

Poznámka:

- Rozmyslete splnění všech požadavků v druhém příkladu!

Příklady

Příklad.

Na množině $\mathbb{N}$ zaveďme relaci $\mathcal{R}$: $n\mathcal{R}m$ kdykoliv $\gcd(n,m)=1$.
Jinak řečeno, n a m jsou spolu v relaci $\mathcal{R}$, právě když jsou nesoudělná.

- $(2,3)\in\mathcal{R}$, tj. $2\mathcal{R}3$ je pravdivé.
- $(2,4)\notin\mathcal{R}$, tj. $2\mathcal{R}4$ není pravdivé.
- Tato relace není relací uspořádání. Proč?

Příklad.

Je-li M množina reálných čísel a $x\mathcal{R}y$ znamená „x je menší nebo rovno y“, pak $\mathcal{R}$ představuje uspořádání na množině $\mathbb{R}$.

Poznámka:

- Rozmyslete splnění všech požadavků v druhém příkladu!
- Toto uspořádání je tzv. **úplné**. Pro každé $x,y\in M$ platí $x\mathcal{R}y$ nebo $y\mathcal{R}x$.

Příklad

Příklad.

Uvažme množinu přirozených čísel $M = \{1, 2, 3, \ldots\}$ a nechť $m\mathcal{R}n$ právě když m dělí n. Tato relace $\mathcal{R}$ na M je uspořádáním.

Příklad

Příklad.

Uvažme množinu přirozených čísel $M = \{1, 2, 3, \ldots\}$ a nechť $m\mathcal{R}n$ právě když m dělí n. Tato relace $\mathcal{R}$ na M je uspořádáním.

Ověřme potřebné vlastnosti

Příklad

Příklad.

Uvažme množinu přirozených čísel $M = \{1, 2, 3, \dots\}$ a nechť $m\mathcal{R}n$ právě když m dělí n. Tato relace $\mathcal{R}$ na M je uspořádáním.

Ověřme potřebné vlastnosti

1. (reflexivita): Pro každé $n \in M$ platí, že n dělí n, tedy $n\mathcal{R}n$.

Příklad

Příklad.

Uvažme množinu přirozených čísel $M = \{1, 2, 3, \ldots\}$ a nechť $m\mathcal{R}n$ právě když m dělí n. Tato relace $\mathcal{R}$ na M je uspořádáním.

Ověřme potřebné vlastnosti

1. (reflexivita): Pro každé $n \in M$ platí, že n dělí n, tedy $n\mathcal{R}n$.
2. (antisymetrie): Uvažme $n, m \in M$ tak, že n dělí m a m dělí n, pak existují $k, \ell \in M$ splňující
$$m = k \cdot n \quad \text{a} \quad n = \ell \cdot m.$$
Tudíž $m = (k\ell) \cdot m$, což může nastat pouze v případě $k = \ell = 1$. Proto $m = n$.

Příklad

Příklad.

Uvažme množinu přirozených čísel $M = \{1, 2, 3, \dots\}$ a nechť $m\mathcal{R}n$ právě když m dělí n. Tato relace $\mathcal{R}$ na M je uspořádáním.

Ověřme potřebné vlastnosti

1. (reflexivita): Pro každé $n \in M$ platí, že n dělí n, tedy $n\mathcal{R}n$.
2. (antisymetrie): Uvažme $n, m \in M$ tak, že n dělí m a m dělí n, pak existují $k, \ell \in M$ splňující
$$m = k \cdot n \quad \text{a} \quad n = \ell \cdot m.$$
Tudíž $m = (k\ell) \cdot m$, což může nastat pouze v případě $k = \ell = 1$. Proto $m = n$.
3. (tranzitivita): Podobně, pokud n dělí m a m dělí k, pak n dělí k.

Příklad

Příklad.

Uvažme množinu přirozených čísel $M = \{1, 2, 3, \ldots\}$ a nechť $m\mathcal{R}n$ právě když m dělí n. Tato relace $\mathcal{R}$ na M je uspořádáním.

Ověřme potřebné vlastnosti

1. (reflexivita): Pro každé $n \in M$ platí, že n dělí n, tedy $n\mathcal{R}n$.
2. (antisymetrie): Uvažme $n, m \in M$ tak, že n dělí m a m dělí n, pak existují $k, \ell \in M$ splňující
$$m = k \cdot n \quad \text{a} \quad n = \ell \cdot m.$$
Tudíž $m = (k\ell) \cdot m$, což může nastat pouze v případě $k = \ell = 1$. Proto $m = n$.
3. (tranzitivita): Podobně, pokud n dělí m a m dělí k, pak n dělí k.

Poznámka:

Toto uspořádání již není úplné. Například pro čísla 2 a 3 není pravda $2\mathcal{R}3$ ani $3\mathcal{R}2$.

Hlavní body

1 Uspořádání

2 Složitost jednoduchých třídících algoritmů

Problém třídění

- **Vstup**: množina $\{x_1, \ldots, x_n\}$ a úplné uspořádání $\leq$ mezi jejími prvky. Velikostí vstupu rozumíme jednoduše počet prvků vstupu.

- **Výstup**: uspořádaná n-tice $(x_{k_1}, \ldots, x_{k_n})$, kde $(k_1, \ldots, k_n)$ je permutace množiny $\{1, \ldots, n\}$ a $x_{k_1} \leq \cdots \leq x_{k_n}$.

- **Elementární operace**: porovnání dvou prvků.

Bublinkový algoritmus (*Bubble sort*)

```
procedure bubbleSort(A : array of length n)
   for k in n-1 to 1 do
      sorted = true
      for i in 1 to k do
         if A[i] > A[i+1] then
            swap( A[i], A[i+1])
            sorted = false
         end if
      end for
      if sorted then return A
   end for
   return A
end procedure
```

Bublinkový algoritmus (*Bubble sort*)

- Celkový počet porovnání může být nejhůře

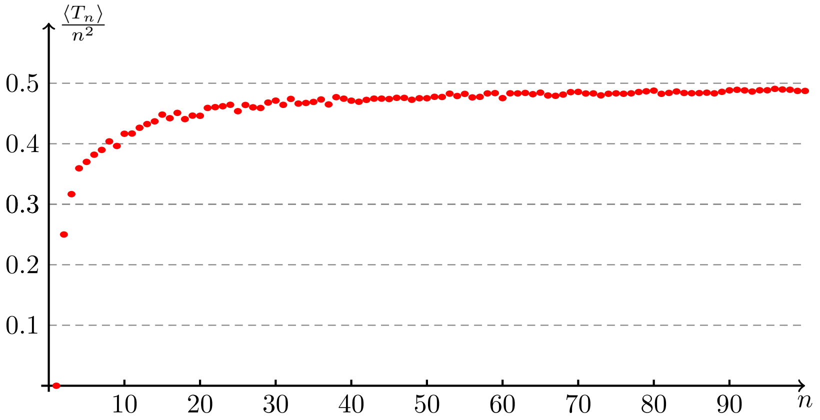
$$(n-1)+(n-2)+\cdots+2+1=\sum_{k=1}^{n-1} k=\frac{n(n-1)}{2}.$$

- Pokud je na vstupu již setříděný seznam, pak algoritmus provede $(n-1)$ porovnání (nejlepší varianta).

Poznámka (Složitost Bubble sort):

Můžeme tedy shrnout, že pro složitost Bubble sort platí

$$T_n=\mathcal{O}(n^2).$$



Poznámka (Ilustrace složitosti Bubble sort):

Pro každé $n \in \{1, 2, \ldots, 100\}$ jsme 20-krát setřídili náhodně permutovanou množinu $\{1, 2, \ldots, n\}$ a vypočetli tak střední hodnotu $\langle T_n \rangle$. Na grafu je pak vynesen poměr $\frac{\langle T_n \rangle}{n^2}$.

Quick sort

Nechť je opět dán vstup $\{x_1, x_2, \ldots, x_n\}$.

Quick sort

Nechť je opět dán vstup $\{x_1, x_2, \ldots, x_n\}$.

- Vyberme náhodně prvek ze seznamu (nazývaný **pivot**).

Quick sort

Nechť je opět dán vstup $\{x_1, x_2, \ldots, x_n\}$.

- Vyberme náhodně prvek ze seznamu (nazývaný **pivot**).
- Prvky ze seznamu menší než pivot, dej do jednoho podseznamu a prvky větší do druhého podseznamu.

Quick sort

Nechť je opět dán vstup $\{x_1, x_2, \ldots, x_n\}$.

- Vyberme náhodně prvek ze seznamu (nazývaný **pivot**).
- Prvky ze seznamu menší než pivot, dej do jednoho podseznamu a prvky větší do druhého podseznamu.
- Dva kratší seznamy uspořádej podle velikosti.

Quick sort

Nechť je opět dán vstup $\{x_1, x_2, \ldots, x_n\}$.

- Vyberme náhodně prvek ze seznamu (nazývaný **pivot**).
- Prvky ze seznamu menší než pivot, dej do jednoho podseznamu a prvky větší do druhého podseznamu.
- Dva kratší seznamy uspořádej podle velikosti.
- Vezmi uspořádaný první seznam, za něj dej pivota a připoj uspořádaný druhý seznam.

Quick sort

Nechť je opět dán vstup $\{x_1, x_2, \ldots, x_n\}$.

- Vyberme náhodně prvek ze seznamu (nazývaný **pivot**).
- Prvky ze seznamu menší než pivot, dej do jednoho podseznamu a prvky větší do druhého podseznamu.
- Dva kratší seznamy uspořádej podle velikosti.
- Vezmi uspořádaný první seznam, za něj dej pivota a připoj uspořádaný druhý seznam.

Algoritmus probíhá rekurentně. Uspořádání dvouprvkového seznamu je jednoduché.

Quick sort

Ozačme nyní T_n **průměrný počet porovnání** pro uspořádání seznamu délky n pomocí algoritmu Quick sort.

Quick sort

Ozačme nyní T_n **průměrný počet porovnání** pro uspořádání seznamu délky n pomocí algoritmu Quick sort.

Pokud je pivot r-tým prvkem seznamu (co do velikosti), pak

$$T_n = n - 1 + T_{r-1} + T_{n-r}, \quad \text{kde klademe } T_0 = T_1 = 0.$$

Quick sort

Ozačme nyní T_n **průměrný počet porovnání** pro uspořádání seznamu délky n pomocí algoritmu Quick sort.

Pokud je pivot r-tým prvkem seznamu (co do velikosti), pak

$$T_n = n - 1 + T_{r-1} + T_{n-r}, \quad \text{kde klademe } T_0 = T_1 = 0.$$

Sečtením těchto vztahů pro $r = 1, 2, \ldots, n$:

$$\sum_{r=1}^{n} T_n = \sum_{r=1}^{n} (n-1) + \sum_{r=1}^{n} \big(T_{r-1} + T_{n-r}\big) \quad \Longrightarrow \quad nT_n = n(n-1) + 2\sum_{r=1}^{n-1} T_r.$$

Quick sort

Ozačme nyní T_n **průměrný počet porovnání** pro uspořádání seznamu délky n pomocí algoritmu Quick sort.

Pokud je pivot r-tým prvkem seznamu (co do velikosti), pak

$$T_n = n - 1 + T_{r-1} + T_{n-r}, \quad \text{kde klademe } T_0 = T_1 = 0.$$

Sečtením těchto vztahů pro $r = 1, 2, \ldots, n$:

$$\sum_{r=1}^{n} T_n = \sum_{r=1}^{n} (n-1) + \sum_{r=1}^{n} \left(T_{r-1} + T_{n-r}\right) \quad \Longrightarrow \quad nT_n = n(n-1) + 2\sum_{r=1}^{n-1} T_r.$$

Poslední rovnost vyjádřeme pro $k-1$ místo k a oba vztahy odečtěme (zbavíme se tím součtu vpravo):

$$kT_k = k(k-1) + 2\sum_{r=1}^{k-1} T_r,$$

$$(k-1)T_{k-1} = (k-1)(k-2) + 2\sum_{r=1}^{k-2} T_r,$$

odečtením: $kT_k - (k-1)T_{k-1} = k(k-1) - (k-1)(k-2) + 2T_{k-1}.$

Quick sort

Odvodili jsme tedy vztah

$$kT_k - (k+1)T_{k-1} = 2k - 2.$$

Quick sort

Odvodili jsme tedy vztah

$$kT_k - (k+1)T_{k-1} = 2k - 2.$$

Vydělením číslem $k(k+1)$ dostáváme

$$\frac{T_k}{k+1} - \frac{T_{k-1}}{k} = \frac{2k-2}{k(k+1)}.$$

Quick sort

Odvodili jsme tedy vztah

$$kT_k - (k+1)T_{k-1} = 2k - 2.$$

Vydělením číslem $k(k+1)$ dostáváme

$$\frac{T_k}{k+1} - \frac{T_{k-1}}{k} = \frac{2k-2}{k(k+1)}.$$

Konečně, sečtením těchto rovností pro $k = 1, 2, \ldots, n$

$$\frac{T_n}{n+1} = 2\sum_{k=1}^{n} \frac{k-1}{k+1} \cdot \frac{1}{k} \leq 2\sum_{k=1}^{n} \frac{1}{k} \leq$$
$$\leq 2\left(1 + \int_1^n \frac{1}{x}\,\mathrm{d}x\right) = 2\big(1 + \ln(n)\big).$$

Uzavíráme

$$T_n = \mathcal{O}\big(n \ln(n)\big).$$