

Strategie řešení rozděl a panuj

doc. Mgr. Jiří Dvorský, Ph.D.

Stav prezentace ke dni 15. září 2024

Katedra informatiky

Fakulta elektrotechniky a informatiky

VŠB – TU Ostrava



Strategie řešení rozděl a panuj

Násobení velkých celých čísel

Strassenovo násobení matic

Problém nejbližší dvojice bodů

Konvexní obal množiny





Strategie řešení rozděl a panuj

Násobení velkých celých čísel

Násobení velkých celých čísel

- Násobení „běžných“ celých čísel řeší procesor.
- Co násobení mnohem větších čísel, se stovkami číslic? Například v kryptografii.
- Určitě by šlo implementovat algoritmus podobný písemnému násobení.
- Jeho provedení vyžaduje n^2 násobení číslic, kde n je počet číslic.

$$\begin{array}{r} 23 \\ 14 \\ \hline 92 \\ 230 \\ \hline 322 \end{array}$$

Otázka k řešení

Lze to provést rychleji? Nebo je toto nejlepší možný algoritmus?

Násobení velkých celých čísel – násobení 23 a 14

Určíme dekadický rozvoj čísel

$$23 = 2 \cdot 10^1 + 3 \cdot 10^0$$

$$14 = 1 \cdot 10^1 + 4 \cdot 10^0$$

A oba rozvoje mezi sebou vynásobíme

$$\begin{aligned} 23 \times 14 &= (2 \cdot 10^1 + 3 \cdot 10^0) \times (1 \cdot 10^1 + 4 \cdot 10^0) \\ &= (2 \times 1) \cdot 10^2 + (2 \times 4 + 3 \times 1) \cdot 10^1 + (3 \times 4) \cdot 10^0 \\ &= 322 \end{aligned}$$

Pro výpočet jsme potřebovali 4 násobení (označena \times), tj. n^2 násobení.

Násobení velkých celých čísel – násobení 23 a 14 (pokrač.)

Prostřední výraz (desítky) lze vyhodnotit i takto

$$2 \times 4 + 3 \times 1 = (2 + 3) \times (1 + 4) - 2 \times 1 - 3 \times 4$$

Neviděli jsme výrazy 2×1 a 3×4 už někde...?

Obecněji mějme $a = a_1 a_0$ a $b = b_1 b_0$ potom

$$c = a \times b = c_2 \cdot 10^2 + c_1 \cdot 10^1 + c_0,$$

kde

- $c_2 = a_1 \times b_1$ je násobek prvních číslic,
- $c_0 = a_0 \times b_0$ je násobek druhých číslic a
- $c_1 = (a_1 + a_0) \times (b_1 + b_0) - (c_2 + c_0)$ je násobek součtů číslic a a b minus číslice c_2 a c_0 .

Násobení velkých celých čísel – rozděl a panuj

Mějme dvě n -ciferná čísla a a b , kde n je sudé přirozené číslo.

Označíme první polovinu číslic čísla a jako a_1 , druhou polovinu jako a_0 . Zápis $a = a_1 a_0$ budeme chápat jako

$$a = a_1 a_0 = a_1 \cdot 10^{n/2} + a_0$$

Pro $b = b_1 b_0$ platí obdobný vztah.

Násobení velkých celých čísel – rozděl a panuj

Součin $c = a \times b$ můžeme zapsat jako

$$\begin{aligned}c &= (a_1 \cdot 10^{n/2} + a_0) \times (b_1 \cdot 10^{n/2} + b_0) \\&= (a_1 \times b_1) \cdot 10^n + (a_1 \times b_0 + a_0 \times b_1) \cdot 10^{n/2} + (a_0 \times b_0) \\&= c_2 \cdot 10^n + c_1 \cdot 10^{n/2} + c_0,\end{aligned}$$

kde

- $c_2 = a_1 \times b_1$ je násobek prvních polovin,
- $c_0 = a_0 \times b_0$ je násobek druhých polovin a
- $c_1 = (a_1 + a_0) \times (b_1 + b_0) - (c_2 + c_0)$ je násobek součtů polovin čísel a a b minus součet c_2 a c_0 .

Čísla c_2, c_1 a c_0 vypočteme stejným způsobem – rekurzivní algoritmus.

Ukončení rekurze: $n = 1$ nebo čísla a, b lze násobit pomocí HW.

Násobení velkých celých čísel – počet násobení

- Počet násobení nutných pro výpočet součinu dvou n -ciferných čísel označíme jako $M(n)$.
- Výpočet součinu vyžaduje 3 násobení čísel poloviční velikosti. Násobení čísel pro $n = 1$ vyžaduje jedno násobení. Tedy

$$M(n) = 3M\left(\frac{n}{2}\right) \text{ pro } n > 1$$

$$M(1) = 1$$

- Metodou zpětné substituce pro $n = 2^k$ dostáváme

$$\begin{aligned}M(2^k) &= 3M(2^{k-1}) = 3[3M(2^{k-2})] = 3^2M(2^{k-2}) \\ &\vdots \\ &= 3^i M(2^{k-i}) \\ &\vdots \\ &= 3^k M(2^{k-k}) = 3^k\end{aligned}$$

- Vzhledem k tomu, že $k = \log_2 n$ dostáváme dále

$$M(n) = 3^{\log_2 n} = n^{\log_2 3} \approx n^{1,585}$$

Poznámky

1. Pro logaritmy platí $a^{\log_b c} = c^{\log_b a}$.
2. Rekurse nemusí nutně pokračovat až k $n = 1$, lze skončit dříve a pro malá n použít běžný algoritmus.

Násobení velkých celých čísel – počet sčítání a odčítání

- Jak je to ale se sčítáním a odčítáním? Není nižší počet násobení vykoupěn vyšším počtem sčítání a násobení?
- Označme $A(n)$ počet sčítání a odčítání při násobení dvou n ciferných čísel.
- Kromě $3A\left(\frac{n}{2}\right)$ operací nutných pro rekurzivní výpočet c_2, c_1 a c_0 potřebujeme 5 součtů a 1 odečítání (na slajdu 341 označena barevně), tedy

$$A(n) = 3A\left(\frac{n}{2}\right) + cn \text{ pro } n > 1$$

$$A(1) = 1$$

- Podle vztahu (??), **Master theorem**, dostáváme

$$A(n) \in \Theta(n^{\log_2 3})$$

- Celkový počet sčítání a odčítání roste asymptoticky stejnou rychlostí jako počet násobení.

- Autorem algoritmu je sovětský matematik Anatolij Alexejevič Karacuba (1937 – 2008), který jej představil v roce 1960.
- Do té doby převládal názor, že tradiční algoritmus je asymptoticky optimální.
- Takže má smysl se zabývat i již „vyřešenými“ problémy :-)
- Otázkou je kdy použít standardní algoritmus a kdy algoritmus založený na strategii rozděl a panuj.

Strategie řešení rozděl a panuj

Strassenovo násobení matic

Strassenovo násobení matic

- Je násobení matic pomocí strategie hrubou silou nejlepší možné?
- Složitost násobení hrubou silou je $\Theta(n^3)$.
- Asymptoticky lepší algoritmus představil Volker Strassen v roce 1969.
- Výchozí „objev“ – násobení čtvercových matic řádu 2 lze provést se 7 násobeními, na rozdíl od 8 u hrubé síly.

Strassenovo násobení matic řádu 2

$$\begin{aligned}\begin{pmatrix} c_{0,0} & c_{0,1} \\ c_{1,0} & c_{1,1} \end{pmatrix} &= \begin{pmatrix} a_{0,0} & a_{0,1} \\ a_{1,0} & a_{1,1} \end{pmatrix} \times \begin{pmatrix} b_{0,0} & b_{0,1} \\ b_{1,0} & b_{1,1} \end{pmatrix} \\ &= \begin{pmatrix} m_1 + m_4 - m_5 + m_7 & m_3 + m_5 \\ m_2 + m_4 & m_1 + m_3 - m_2 + m_6 \end{pmatrix}\end{aligned}$$

$$m_1 = (a_{0,0} + a_{1,1})(b_{0,0} + b_{1,1})$$

$$m_5 = (a_{0,0} + a_{0,1})b_{1,1}$$

$$m_2 = (a_{1,0} + a_{1,1})b_{0,0}$$

$$m_6 = (a_{1,0} - a_{0,0})(b_{0,0} + b_{0,1})$$

$$m_3 = a_{0,0}(b_{0,1} - b_{1,1})$$

$$m_7 = (a_{0,1} - a_{1,1})(b_{1,0} + b_{1,1})$$

$$m_4 = a_{1,1}(b_{1,0} - b_{0,0})$$

Strassenovo násobení matic

- Počty operací pro matice 2×2 :

	Hrubá síla	Strassen
Počet násobení	8	7
Počet sčítání a odčítání	4	18

- Násobit takto matice 2×2 je očividný nesmysl. Ale!

Strassenovo násobení matic (pokrač.)

- Vztahy můžeme přeformulovat na převod násobení matic $n \times n$ na podmatice řádu $\frac{n}{2} \times \frac{n}{2}$ takto:

$$\left(\begin{array}{c|c} C_{0,0} & C_{0,1} \\ \hline C_{1,0} & C_{1,1} \end{array} \right) = \left(\begin{array}{c|c} A_{0,0} & A_{0,1} \\ \hline A_{1,0} & A_{1,1} \end{array} \right) \times \left(\begin{array}{c|c} B_{0,0} & B_{0,1} \\ \hline B_{1,0} & B_{1,1} \end{array} \right)$$

- Podmatici $C_{0,0}$ můžeme vypočítat buď jako

$$C_{0,0} = A_{0,0} \times B_{0,0} + A_{0,1} \times B_{1,0}$$

nebo jako

$$C_{0,0} = M_1 + M_4 - M_5 + M_7$$

- Matice M_1, \dots, M_7 vypočteme stejným, rekurzivním, způsobem.

Strassenovo násobení matic – analýza složitosti

Počet násobení $M(n)$ pro matice $n \times n$ je dán rekurentní rovnicí:

$$M(n) = 7M\left(\frac{n}{2}\right) \text{ pro } n > 1$$

$$M(1) = 1$$

Předpokládejme, že $n = 2^k$ a odtud dostáváme

$$M(2^k) = 7M(2^{k-1}) = 7[7M(2^{k-2})] = 7^2M(2^{k-2})$$

$$\vdots$$

$$= 7^i M(2^{k-i})$$

$$\vdots$$

$$= 7^k M(2^{k-k}) = 7^k.$$

Protože $k = \log_2 n$ a tudíž

$$M(n) = 7^{\log_2 n} = n^{\log_2 7} \approx n^{2.807} < n^3$$

Strassenovo násobení matic – analýza složitosti, sčítání

- Neroste ale počet sčítání $A(n)$ pro matice $n \times n$ příliš rychle?
- Pro násobení matic $n \times n$ potřebujeme:
 1. vypočítat 7 podmatic řádu $\frac{n}{2} \times \frac{n}{2}$ a
 2. provést 18 sčítání/odečítání podmatic řádu $\frac{n}{2} \times \frac{n}{2}$.

Takže

$$A(n) = 7A\left(\frac{n}{2}\right) + 18\left(\frac{n}{2}\right)^2 \text{ pro } n > 1$$
$$A(1) = 0$$

- Podle vztahu (??), **Master theorem**, dostáváme

$$A(n) \in \Theta\left(n^{\log_2 7}\right)$$

- Z toho plyne, že Strassenovo násobení matic má asymptotickou složitost $\Theta\left(n^{\log_2 7}\right)$, což je méně než řešení hrubou silou.

Strategie řešení rozděl a panuj

Problém nejbližší dvojice bodů

Strategie řešení rozděl a panuj

Konvexní obal množiny

Děkuji za pozornost

1. **Železniční mapy ČR** [online]. Praha: SŽDC, 2019 [cit. 2019-11-15]. Dostupné z: *<http://provoz.szdc.cz/portal/Show.aspx?path=/Data/Mapy/kjr.pdf>*.
2. LEVITIN, Anany. **Introduction to the Design and Analysis of Algorithms**. 3rd ed. Boston: Pearson, 2012. ISBN 978-0-13-231681-1.
3. CORMEN, Thomas H.; LEISERSON, Charles Eric; RIVEST, Ronald L.; STEIN, Clifford. **Introduction to algorithms**. Fourth edition. Cambridge, Massachusetts: The MIT Press, 2022. ISBN 978-026-2046-305.

4. WRÓBLEWSKI, Piotr. **Algoritmy**. 1. vyd. Brno: Computer Press, 2015. ISBN 978-80-251-4126-7.