

# Základy algoritmizace

Matematické algoritmy (11MAG)

Jan Příkryl

1. přednáška 11MAG  
pondělí 30. září 2013

verze: 2013-10-22 14:30

## Obsah

|                                    |          |
|------------------------------------|----------|
| <b>1 Úvod</b>                      | <b>1</b> |
| <b>2 Algoritmy a algoritmizace</b> | <b>2</b> |
| 2.1 Vlastnosti algoritmů . . . . . | 2        |
| <b>3 Aplikace</b>                  | <b>3</b> |
| <b>4 Prerekvizity předmětu</b>     | <b>5</b> |

## 1 Úvodní informace

**Přednáší:** Dr. Jan Příkryl <prikryl@fd.cvut.cz>

**Přednáška:** pondělí, 16:45–18:15, F114

**Cvičení:** pondělí 18:30–20:00, F311  
úterý 16:45–18:15, F311

**Dotace:** 2+2

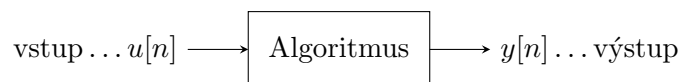
**Očekávaná týdenní zátěž:** 5–7 hodin včetně přednášek

Webové stránky předmětu: <http://zolotarev.fd.cvut.cz/mag/>

**O čem si budeme povídat:** algoritmy diskrétní matematiky, slasti a strasti výpočtů v plovoucí řádové čárce, numerická matematika.

**O čem budou cvičení:** praktické hrátky s algoritmy, Matlab/Python/C/C++/Java . . .

**Co když neumím programovat?** To, že jste postoupili až do prvního magisterského ročníku garantuje, že programovat umíte. V případě nouze se urychleně doučíte. Učebnic základů algoritmizace a programování existuje celá řada, můžete zkusit třeba [12, 5] či informace na stránkách



**Obrázek 1:** Formální pohled na algoritmus ze systémového hlediska

Katedry počítačů ČVUT FEL [1], například ty o předmětu Algoritmizace [8].

**Podle čeho se to učí:** Literatura je téměř výhradně anglicky, kompletní seznam i s případnými odkazy naleznete na webových stránkách předmětu a také na konci tohoto textu.

## 2 Algoritmy a algoritmizace

Algoritmus je

- přesný návod či postup, kterým lze vyřešit daný typ úlohy.
- efektivní postup pro výpočet hodnoty nějaké funkce vyjádřený konečným počtem instrukcí.

**Definice 1.** Algoritmem rozumíme postup, podle kterého se z dat vstupních  $x[n]$  vygenerují data výstupní  $y[n]$ .

- Typy algoritmů
- Co potřebujete znát ?
- Kam až můžeme dojít ?

### 2.1 Vlastnosti algoritmů

Každý algoritmus musí mít následující vlastnosti:

1. **Konečnost:** výpočet se ukončí v „rozumně“ konečném čase.
2. **Hromadnost:** není sestaven pouze na jediné  $u[n]$ , ale na celou řadu možných vstupů.
3. **Jednoznačnost:** přechod do následujícího stavu algoritmu je jednoznačně určen výsledkem stavu předchozího.

#### Komentář k vlastnostem algoritmů

1. **Konečnost:** předpověď počasí na zítra dosažená výpočtem o den později nemá význam.
2. **Hromadnost:** program pro výpočet odmocniny pracuje nad množinou čísel, není konstruován pro každé číslo zvlášť.
3. **Jednoznačnost:** každý algoritmus je složen z kroků, které na sebe vzájemně navazují. Každý krok je charakterizován jako přechod z jednoho stavu do jiného. Každý stav algoritmu je určen zpracovávanými daty a na tom, jak data v jednotlivých stavech vypadají. Je tedy pevně určeno, který krok bude následovat.

Další variantou je numerický výpočet druhé odmocniny čísla  $x$  pomocí nelineární diferenční rovnice

$$y[n+1] = \frac{1}{2} \left( y[n] + \frac{u[n]}{y[n]} \right),$$

kde vstupní posloupnost  $u[n] = x \cdot \mathcal{K}[n]$  a výstupní posloupnost  $y[n]$  postupně konverguje k hodnotě odmocniny. Počáteční podmínku můžeme volit v podstatě libovolnou, například  $y[0] = x$  nebo  $y[0] = 1$ .

Odmocnina z čísla 10 je s přesností na 10 desetinných míst rovna  $\sqrt{10} = 3,16227766017$ .

Pro  $u[n] = 10$  dostáváme postupně

$$\begin{array}{ll} y[0] = 3 & y[0]^2 = 9 \\ y[1] = 3,165 & y[1]^2 = 10,017225 \\ y[2] = 3,162278 & y[2]^2 = 10,0000021493 \\ y[3] = 3,1622776601 & y[3]^2 = 9,9999999996 \\ \vdots & \end{array}$$

### 3 Aplikace algoritmů

Internet umožňuje lidem na celém světě rychle vyhledávat a přistupovat k obrovskému množství informací. Aby to fungovalo, musí poskytovatelé internetu a poskytovatelé internetových služeb používat chytré algoritmy, umožňující zpracovávat a spravovat tak velké množství dat. Příklady úloh, na které v této oblasti narazíme, jsou například *hledání vhodných cest* pro datové pakety, cestující mezi jednotlivými uzly sítě, či *vyhledávání stránek* s určitým obsahem.

Velký objem obchodů je v dnešní době uzavírán elektronicky, a mnoho služeb funguje i na elektronické bázi. Pro celý obor e-komerce je zcela klíčová schopnost uchovat důvěrné údaje (čísla kreditních karet, hesla, či bankovní informace) opravdu v tajnosti. Mezi úlohy z této oblasti, na které narazíme, patří *šifrování veřejným klíčem* či *digitální podpis*.

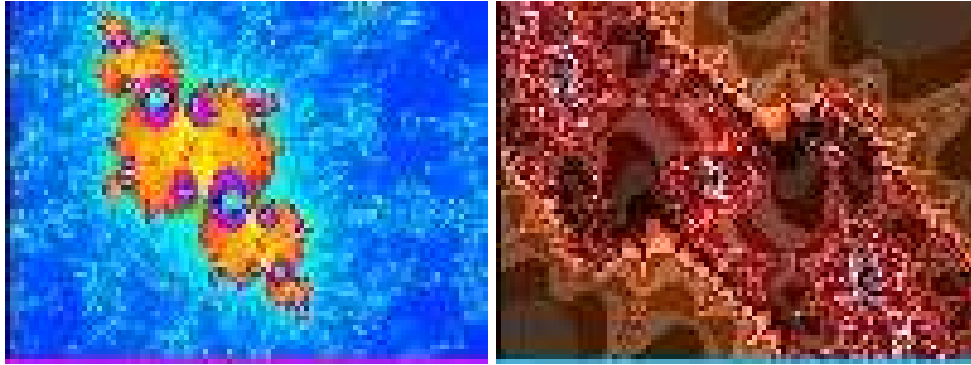
Ve oblastech výroby a přepravy řeší firmy často problém optimální alokace zdrojů tak, aby byly na jednu stranu minimalizovány výrobní či režijní náklady, na druhou stranu aby byl co možná nejvyšší užitek. Letecký dopravce se bude snažit přiřazovat posádky na jednotlivé lety způsobem, jenž generuje co nejmenší dodatečné náklady – zároveň je jeho manévrovací prostor ovšem omezen nařízením z oblasti bezpečnosti provozu, všeobecnými právními předpisy a podobně. Poskytovatel internetu při investicích do infrastruktury potřebuje investovat své zdroje cíleně tak, aby výsledek co nejefektivněji sloužil zákazníkům. Obě tyto úlohy řeší algoritmy matematické optimalizační techniky zvané *lineární programování*.

Proč to zkoumat?

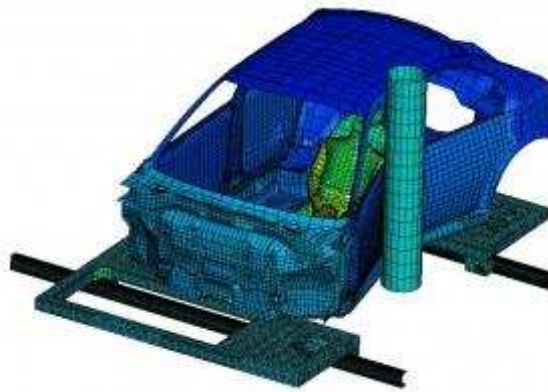
Numerické řešení algebraických rovnic, diferenciálních rovnic a speciálních funkcí:

Metoda konečných prvků – řešení složitých parciálních diferenciálních rovnic s praktickými aplikacemi:

Jinak těžko řešitelné úlohy: nelineární parciální diferenciální rovnice, například Navierovy-Stokesovy rovnice. Tyto rovnice popisují proudění a počítáme je například při studiu obtékání vzduchu okolo křidel, viz. Obrázek 4.



**Obrázek 2:** Fraktály (vlevo tzv. Juliina množina, vpravo).



**Obrázek 3:** Metoda konečných prvků.



**Obrázek 4:** I tento jev, způsobený prouděním okolo křídla letounu, lze popsat Navierovými-Stoeksovými rovnicemi.

Navierovy-Stokesovy rovnice:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \nabla) \mathbf{u} = \mathbf{f} - \nabla p + \nu \Delta \mathbf{u}$$

kde  $\mathbf{u}$  a  $\mathbf{f}$  jsou vektorové funkce rychlosti a síly,  $p$  je tlak a  $\nu$  je úměrná viskozitě kapaliny.

$$\begin{aligned} \frac{\partial u_x}{\partial t} + u_x \frac{\partial u_x}{\partial x} + u_y \frac{\partial u_y}{\partial y} + u_z \frac{\partial u_z}{\partial z} = \\ = f_x(x, y, z, t) - \frac{\partial p}{\partial x} + \nu \left[ \frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_x}{\partial y^2} + \frac{\partial^2 u_x}{\partial z^2} \right] \end{aligned}$$

Výše uvedené seznamy nejsou zdaleka vyčerpávající, můžeme z nich ale odvodit dvě základní charakteristiky, společné mnoha zajímavým algoritmům:

1. Pro daný problém existuje většinou velké množství možných řešení, z nichž většina z různých důvodů není to, co potřebujeme.
2. Algoritmy mají praktický užitek. Metoda hledání nejkratší cesty v grafu umožní přepravní společnosti snížit přepravní náklady, neboť sníží náklady na palivo a na práci personálu. Ten samý algoritmus může ve směrovači počítačové sítě hledat způsob, jak co nejrychleji doručit vaši zprávu adresátovi.

### Algoritmy jsou technologie

Výpočetní čas je omezený zdroj, stejně tak, jak je omezená velikost operační paměti počítače. Oba tyto zdroje je třeba využívat rozumně, – algoritmy, jež jsou efektivní z hlediska doby výpočtu či nároků na operační paměť, jsou nástrojem k takovému rozumnému využití.

Algoritmy, stejně jako počítačový hardware, lze v dnešní době považovat za technologii. Celkový výkon systémů závisí na volbě efektivních algoritmů do stejné míry, jako závisí na volbě dostatečně výkonného hardware. A s tím, jak se vyvíjí počítačové technologie, se ruku v ruce vyvíjí i algoritmy.

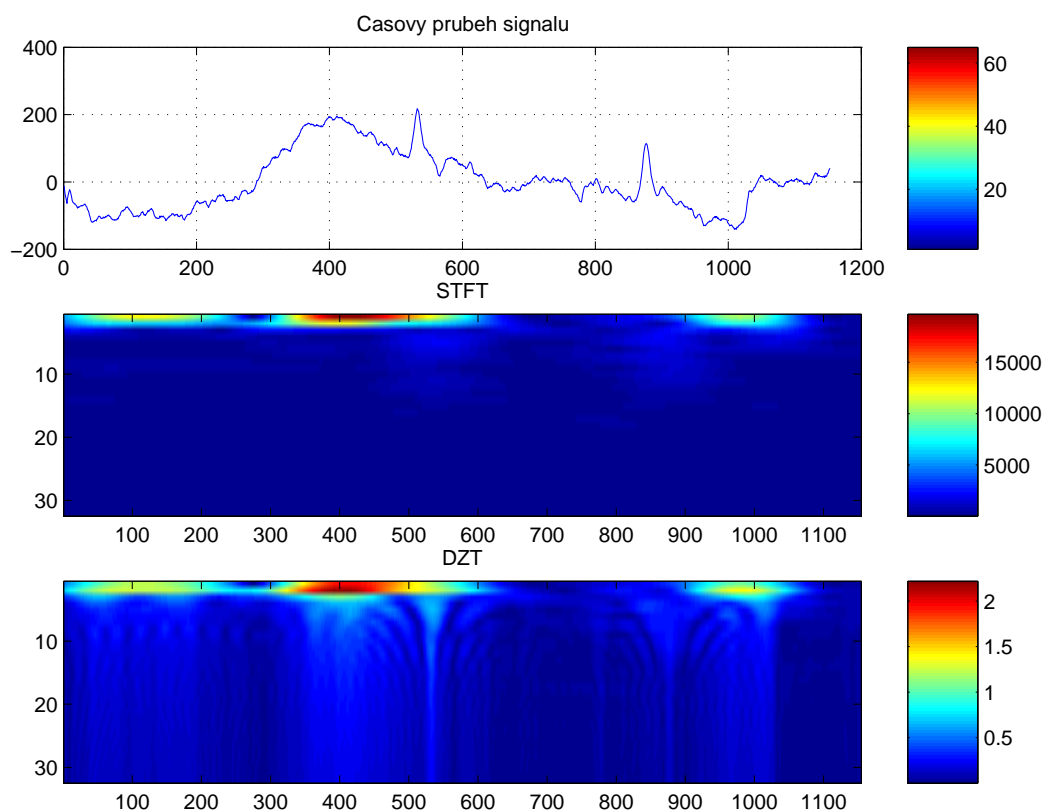
## 4 Prerekvizity předmětu

Předpokládáme:

- základy algebry a matematické analýzy
- základy numerické matematiky
- diferenční rovnice a jejich řešení
- základy strukturovaného programování
- aktivní znalost alespoň jednoho programovacího jazyka (C, C++, Python, Java, Basic) nebo alespoň prostředí MATLAB

Kam až můžeme dojít?

## Rychlá Fourierova transformace – analýza EEG signálu



**Obrázek 5:** Rychlá Fourierova transformace – analýza EEG signálu

- Objevit krásu některých algoritmů.
- Pochopit třeba numerické základy kryptologie.
- Nebát se inženýrských úloh, které vyžadují algoritmizaci.
- Pochopit rychlé algoritmy s aplikacemi v reálném světě

Kurs pokrývá standardní algoritmy, jež nabízí pro daný problém a pro daná vstupní data optimální výkon.

Dvě nejčastější chyby při výběru algoritmu pro danou úlohu:

- **ignorujeme výkon algoritmu** – rychlejší algoritmy jsou současně složitější na implementaci
- **příliš zkoumáme výkon algoritmu** – nepatrně rychlejší algoritmus může být výrazně složitější na implementaci

## Reference

- [1] Katedra počítačů ČVUT FEL.
- [2] Thomas H. Cormen, Charles E. Leiserson, R.L. Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press / McGraw-Hill, 2 edition, 2002.
- [3] Johan Håstad. *Advanced Algorithms - Course Notes*. Royal Institute of Technology, Sweden, 2000.
- [4] Michael T. Heath. *Scientific computing: An introductory Survey*. McGraw-Hill, 2 edition, 2002.
- [5] Bohuslav Hudec. *Programovací techniky*. Skripta. Česká technika – nakladatelství ČVUT, 2001.
- [6] Malvin H. Kalos and Paula A. Whitlock. *Monte Carlo Methods*. Wiley-Interscience, 1986.
- [7] David J.C. Mackay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.
- [8] Jiří Zdeněk Miroslav Balík, Božena Mannová. Algoritmizace (x36alg) – podklady k přednáškám.
- [9] Cleve B. Moler. *Numerical computing with MATLAB*. Society for Industrial Mathematics (SIAM), 2004.
- [10] Hans-Görg Roos and Hubert Schwitlick. *Numerische Mathematik – Das Grundwissen für jedermann*. Mathematik für Ingenieure und Naturwissenschaftler. Teubner, Stuttgart/Leipzig, Stuttgart/Leipzig, 1999.
- [11] Jiří Velebil. Diskrétní matematika. www, 2007. Text k přednášce.
- [12] Miroslav Virius. *Základy algoritmizace*. Skripta. Česká technika – nakladatelství ČVUT, 2008.