

# Základy algoritmizace

## Matematické algoritmy (11MAG)

Jan Prikryl

Ústav aplikované matematiky  
ČVUT v Praze, Fakulta dopravní

1. přednáška 11MAG  
pondělí 5. října 2014

verze: 2014-10-06 11:27



# Obsah přednášky

- 1 Úvodní informace
- 2 Algoritmy a algoritmizace
- 3 Aplikace algoritmů



# Základní informace

Kdo to učí, kdy se uvidíme, web . . .

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

**Přednáška:** pondělí, 11:30–13:00, F114

**Cvičení:** pondělí 13:15–14:45, F215

**Dotace:** 2+2

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

Webové stránky předmětu:

<http://zolotarev.fd.cvut.cz/mag/>



# Základní informace

## Hodnocení

Teoretické testy	3 × 3 b	9 bodů
Práce na cvičení	11 × 1 b	11 bodů
Celkem		20 bodů

A	B	C	D	E	F
20–18	18–16	16–14	14–12	12–10	<10



# Základní informace

O čem to je, co se po mně bude chtít?

**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.



# Základní informace

Podle čeho se to učí

Literatura je téměř výhradně anglicky, kompletní seznam i s případnými odkazy naleznete na webových stránkách předmětu.

- 1 HÅSTAD, Johan: *Advanced Algorithms – Course Notes*. Royal Institute of Technology, Sweden, 2000.
- 2 CORMEN, Thomas H. – LEISERSON, Charles E. – RIVEST, Ronald L. – STEIN, Clifford: *Introduction to Algorithms*. 2nd edition. The MIT Press / McGraw-Hill, 2002.
- 3 MOLER, Cleve: *Numerical Computing with MATLAB*. Society for Industrial and Applied Mathematics (SIAM), 2004.
- 4 HEATH, Michael T.: *Scientific Computing: An Introductory Survey*. 2nd edition. McGraw-Hill, 2002.
- 5 PŘIKRYL, Jan – PŘIKRYL, Petr: *Matematické algoritmy*. ČVUT FD, 2014.



# Obsah přednášky

- 1 Úvodní informace
- 2 Algoritmy a algoritmizace**
- 3 Aplikace algoritmů



# Co víte o algoritmech?

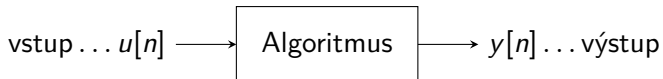
## Co je to algoritmus

Algoritmus je

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

### Definice

Algotmem rozumíme postup, podle kterého se z dat vstupních  $x[n]$  vygenerují data výstupní  $y[n]$ .





# Co víte o algoritmech?

## Vlastnosti algoritmu

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

Každý algoritmus musí mít následující vlastnosti:

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



# Co víte o algoritmech?

## Komentář k vlastnostem algoritmů

- ❶ **Konečnost:** předpověď počasí na zítra dosažená výpočtem o den později nemá význam.
- ❷ **Hromadnost:** program pro výpočet odmocniny pracuje nad množinou čísel, není konstruován pro každé číslo zvlášť.
- ❸ **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.



# Příklady algoritmů

## Příklad

Numerický výpočet odmocniny

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

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

Pro  $u[n] = 10$  dostáváme postupně

$$y[0] = 3$$

$$y[0]^2 = 9$$

$$y[1] = 3,165$$

$$y[1]^2 = 10,017225$$

$$y[2] = 3,162278$$

$$y[2]^2 = 10,0000021493$$

$$y[3] = 3,1622776601$$

$$y[3]^2 = 9,9999999996$$

⋮



# Obsah přednášky

- 1 Úvodní informace
- 2 Algoritmy a algoritmizace
- 3 Aplikace algoritmů**



# Aplikace algoritmů

## Internet

Internet umožňuje lidem na celém světě rychle vyhledávat a přistupovat k obrovskému množství informací.

Poskytovatelé internetu a internetových služeb musí používat chytré algoritmy, umožňující **zpracovávat a spravovat velké množství dat**.

Příklady algoritmů:

- *hledání vhodných cest* pro datové pakety, cestující mezi jednotlivými uzly sítě,
- *vyhledávání* stránek s určitým obsahem.



# Aplikace algoritmů

## Elektronické obchodování

Velký objem obchodů je v dnešní době uzavírán elektronicky, a mnoho služeb funguje i na elektronické bázi.

**E-komerce:** klíčová je schopnost uchovat důvěrné údaje (čísla kreditních karet, hesla, či bankovní informace) opravdu v tajnosti.

Příklady algoritmů:

- *šifrování veřejným klíčem* či
- *digitální podpis*.



# Aplikace algoritmů

## Optimalizace a logistika

Ve oblastech výroby a přepravy řeší firmy často problém optimální alokace zdrojů: minimalizace nákladů vs. co možná nejvyšší užitek.

**Letecký dopravce:** přiřazení posádek na lety s co nejmenšími náklady, optimální využití strojů.

**Poskytovatel internetu:** cílené investice do infrastruktury.

**Svoz odpadu:** Minimum najetých kilometrů.

Příklady algoritmů:

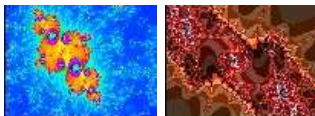
- *lineární* či *dynamické programování* – optimalizace,
- *grafové algoritmy* – komponenty grafu, kostra, nejkratší cesta.



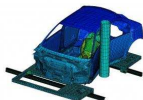
# Aplikace algoritmů

## Numerická matematika

**Numerické řešení** soustav algebraických rovnic, diferenciálních rovnic a speciálních funkcí:



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





# Aplikace algoritmů

## Proč to zkoumat

Jinak těžko řešitelné úlohy: nelineární parciální diferenciální rovnice, například Navierovy-Stokesovy rovnice.



# Aplikace algoritmů

## Proč to zkoumat

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.

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



# Co potřebujete znát?

## Prerekvizity

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?

- Objevit krásu některých algoritmů.
- Pochopit třeba numerické základy kryptologie.
- Nebát se inženýrských úloh, které vyžadují algoritmizaci.

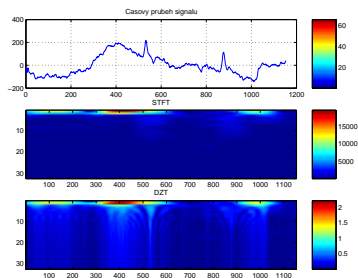


# Kam až můžeme dojít?

pokračování

- Pochopit rychlé algoritmy s aplikacemi v reálném světě

## Rychlá Fourierova transformace – analýza EEG signálu



# Implementace algoritmů

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

