Základy algoritmizace

Matematické algoritmy (11MAG)

Jan Přikryl

1. přednáška 11MAG
úterý 25. září 2012

1 Úvodní informace

Přednáší: Dr. Jan Přikryl ⟨prikryl@fd.cvut.cz⟩ **Přednáška:** úterý, 16:45–18:15, F112 **Cvičení:** úterý 18:30–20:00, 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 do příštího týdne naučíte.

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

1. HÅSTAD, Johan: *Advanced Algoritms – Course Notes.* Royal Institute of Technology, Sweden, 2000.

2. MOLER, Cleve: *Numerical Computing with MATLAB.* Society for Industrial and Applied Mathematics (SIAM), 2004.

3. CORMEN, Thomas H. – LEISERSON, Charles E. – RIVEST, Ronald L. – STEIN, Clifford: *Introduction to Algorithms.* 2nd edition. The MIT Press / McGraw-Hill, 2002.

4. HEATH, Michael T.: *Scientific Computing: An Introductory Survey.* 2nd edition. McGraw-Hill, 2002.

2 Algoritmy a algoritmizace

Definice 1 (podle Wikipedie).

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

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

Definice 2. Algoritmem rozumíme postup, podle kterého se z dat vstupních $x(n)$ vygenerují data výstupní $y(n)$.

$x(n)$ Algoritmus $y(n)$

Každý algoritmus musí mít následující vlastnosti:

1. **Konečnost:** výpočet se ukončí v „rozumně" konečném čase.
2. **Hromadnost:** není sestrojen pouze na jediné $x(n)$, ale na celou řadu možných vstupů.
3. **Jednoznačnost:** přechod do následujícího stavu algoritmu je jednoznačně určen výsledkem stavu předchozího.

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.

3 Příklady algoritmů

Metoda nalezení **největšího společného dělitele** (NSD≡ GCD Greatest Common Divisor) byla známá již ve starověkém Řecku před 2000 lety. Spočívá v jednoduchém pozorování, že největší společný dělitel dvou čísel $p > q$ je shodný s největším společným dělitelem čísel $p-q, q$.

Tento poznatek již stačí k sestavení algoritmu.

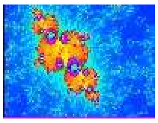
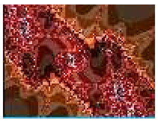
```
given p, q
body
    do if p < q
        r = p;  p = q;  q = r;
    end
        p = p − q;
    repeat until p = 0
result
  gcd =  q
```

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

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

Pro $x(n-1) \equiv x(0) = 10$ dostáváme postupně

$$\begin{array}{ll}
y(1) = 3 & y^2(1) = 9 \\
y(2) = 3.165 & y^2(2) = 10.017225 \\
y(3) = 3.162278 & y^2(3) = 10.0000021493 \\
y(4) = 3.1622776601 & y^2(4) = 9.9999999996 \\
\vdots &
\end{array}$$

Numerické řešení algebraických rovnic, diferenciálních rovnic a speciálních funkcí:  

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

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

Navierovy-Stokesovy rovnice:

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

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

$$\frac{\partial u_x}{\partial t} + u_x \frac{\partial u_x}{\partial x} + u_y \frac{\partial u_y}{\partial y} + u_z \frac{\partial u_z}{\partial z} =$$

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

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

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

- Pochopit rychlé algoritmy s aplikacemi v reálném světě

Rychlá Fourierova transformace – analýza EEG signálu