Předmluva

Počítačová algebra se zabývá algoritmy pro symbolické výpočty v různých algebraických oborech, jako jsou např. celá a racionální čísla, konečná tělesa a především polynomy nad těmito obory, případně další struktury jako algebraická rozšíření racionálních čísel, permutační grupy apod.

Počítačová algebra se svým přístupem zásadním způsobem liší od *numerické matematiky*, která se soustředí především na *přibližné* výpočty, zpravidla nad nějakým počítačovým modelem reálných či komplexních čísel; její důležitou součástí je teorie odhadu chyby. Oproti tomu počítačová algebra vyvíjí algoritmy, které počítají symbolicky, tedy absolutně *přesně*. (Speciálně to znamená, že se nemůže zabývat reálnými čísly, vždy jen nějakým spočetným podoborem.)

Oba přístupy, aproximativní i symbolický, mají svoje místo: různé úlohy vyžadují různý přístup. Uvažme například výpočet kořenů daného polynomu. Z Galoisovy teorie plyne, že symbolicky lze řešit pouze polynomiální rovnice stupně ≤ 4, a to navíc dosti komplikovaně, zato numerická Newtonova metoda umí efektivně řešit rovnice libovolného stupně. Na druhou stranu, u soustav polynomiálních rovnic více proměnných se může vyplatit soustavu upravit na hezčí tvar (podobně jako lineární rovnice Gaussovou eliminací) a zde má místo počítačová algebra. Navíc je řada problémů, které ani nedává smysl řešit přibližně, např. největší společný dělitel dvou polynomů nebo operace nad konečnými tělesy, a těmi především se zabývá počítačová algebra.

Tato učebnice se orientuje především na algoritmy pro počítání s polynomy. Je to jeden z nejdůležitějších směrů počítačové algebry, na který řada dalších problematik navazuje (např. algebraická rozšíření těles se reprezentují pomocí polynomů) a který má řadu aplikací. Prvních pět kapitol se věnuje především polynomům jedné proměnné, šestá kapitola se věnuje jednomu ze základních nástrojů pro práci s polynomy více proměnných a z tématu trochu vybočuje poslední kapitola věnovaná počítání s diskrétními podgrupami vektorových prostorů.

Mezi hlavní oblasti využití algoritmů, které se naučíme, patří výpočetní teorie čísel (testy prvočíselnosti, faktorizace apod.), výpočetní geometrie (algoritmy pro práci s algebraickými křivkami a plochami), kryptografie (návrhy moderních kryptosystémů a především efektivní kryptoanalýza), či složitější algoritmy počítačové algebry, jako je symbolická integrace nebo symbolické řešení diferenciálních rovnic.

Obsah učebnice by se dal shrnout takto: nejprve se seznámíme s datovou reprezentací a rychlými algoritmy pro základní operace (sčítání, násobení, dělení) s čísly a polynomy. V dalších kapitolách se budeme zabývat složitějšími úlohami jako je největší společný dělitel a rozklad na ireducibilní činitele. Mnoho rychlých algoritmů je založeno na tzv. *modulární reprezentaci*, kterou představíme ve druhé kapitole: jejím principem je trik, že místo jednoho výpočtu v komplikovaném oboru provedeme úlohu v několika jednodušších oborech. V některých případech dochází k výrazné úspoře času. Poslední dvě kapitoly jsou věnovány dvěma zajímavým metodám s řadou aplikací: Gröbnerovy báze, které umožňují efektivní manipulaci s polynomy více proměnných, včetně řešení soustav polynomiálních rovnic, a Lenstra-Lenstra-Lovászův algoritmus pro hledání krátkých vektorů v mřížích, na který lze převést řadu problémů ze zdánlivě nesouvisejících oblastí matematiky.

Výklad je doplněn řadou cvičení, která jsou zejména ze začátku výrazně orientována na výpočet odhadu časové složitosti algoritmů. Cvičení občas obsahují alternativní algoritmy řešící úlohy popisované v textu, část cvičení slouží k vyzkoušení

běhu složitějších algoritmů na konkrétních příkladech. Cvičení, která považujeme za zvláště důležitá, mají dvakrát podtržené číslo; důrazně doporučujeme se na ně podívat. Hvězdičkou jsou označena těžší cvičení.

Probírané algoritmy jsou samozřejmě součástí velkých softwarových balíků pro počítačovou algebru, pro programátory je k dispozici řada knihoven v prakticky jakémkoliv jazyce. Pro implementaci s důrazem na efektivitu se v současné době nejvíce používá jazyk C++, s využitím knihoven GMP pro počítání s velkými čísly a NTL pro polynomiální aritmetiku. Dá se říci, že sekce 4 seznamuje s jednoduššími verzemi algoritmů, které obsahuje knihovna GMP, zatímco algoritmy z kapitol II–V a VII tvoří základ knihovny NTL. Pro polynomy více proměnných je v současné době nejefektivnějším volně dostupným nástrojem systém Sage (využívající projekt Singular). Podrobnější informace najdete v sekci 2.

Při přípravě této učebnice jsme vycházeli především z následujících tří monografií:

- Franz Winkler, *Polynomial Algorithms in Computer Algebra*
- K. Geddes, S. Czapor, G. Labahn, *Algorithms for Computer Algebra*
- J. von zur Gathen, J. Gerhard, *Modern Computer Algebra*

Tyto knihy pokrývají větší paletu témat, než bylo účelné zpracovat do naší učebnice, čtenáři zde tedy mohou najít řadu rozšíření a aplikací probírané látky. Kromě výše uvedených knih jsme použili řadu článků a internetových zdrojů, abychom zařadili aktuální informace, případně doplnili některé chybějící detaily.

Kromě výše uvedených knih lze jako doplňující čtení doporučit učebnici Henri Cohena *A Course in Computational Algebraic Number Theory*, která obsahuje především témata týkající se výpočetní teorie čísel, případně slavnou sadu učebnic Donalda Knutha *The Art of Computer Programming*, jejíž jeden díl je věnován algebraickým algoritmům. Na konci každé sekce pak uvádíme odkazy na zdroje specifické k danému tématu.

K obsahu tohoto textu patří následující důležitá poznámka: zatímco např. v matematické analýze se učí poznatky staré řádově stovky let a v základech algebry řádově sto let, v tomto kurzu se probírají poznatky staré řádově desítky či jednotky let. Zatímco většina fakt z analýzy či obecné algebry bude užitečných navěky a s velkou pravděpodobností se je budou učit studenti i za sto let v téměř nezměněné podobě, věhlas a užitečnost algoritmů uvedených v této učebnici stojí a padá s objevem algoritmu rychlejšího. Je dobré si uvědomit, že v celém kurzu dokazujeme pouze horní odhady složitosti. Neuvidíte ani jeden důkaz, který by říkal, že předváděný algoritmus je optimální. Je pravděpodobné, že už za pár let bude tento text zčásti zastaralý.

Na závěr patří *poděkování* všem, kteří se podíleli na vzniku těchto skript. Na počátku významně pomohl student Ivan Štubňa, který přepisoval zápisky z přednášek do elektronické formy. Velké poděkování patří všem studentům, kteří nás upozornili na řadu drobných chyb v předběžných verzích tohoto textu. Za krásnou grafiku na obálce i v textu patří poděkování Evě Stanovské.

V Praze, v červenci 2010,
David Stanovský a Libor Barto