

Oblikovanje i analiza algoritama (akad. god. 2012./13.)

Domaće zadaće

Upute ili “pravila igre”:

- Za zadaću se bira 1 (**jedan**) od ponuđenih 5 zadataka.
- Rješenje je program koji radi to što treba, s popratnom dokumentacijom (ne mora je biti, ali dozvoljeno je, na primjer, sažeto prikazati rezultate opširnijeg testiranja).
- Zadnji rok za predaju zadaće je **ponedjeljak, 28. siječnja 2013., do 14 sati**.

Kodeks ponašanja: Tuđa rješenja (autor programa je netko drugi), ista ili vrlo slična rješenja se **poništavaju**. To se može dogoditi i naknadno, tj. retroaktivno, ako se kasnije pojave “kopije”!

Napomena. Programski zadaci su **namjerno** sažeto formulirani, tako da propisuju samo nužni dio posla. Sve ostalo: planiranje ulaza, izlaza, izbora raznih algoritama, implementacije, testiranja i sl., je “slobodno” i nagrađuje se kao dio rješenja. Posebno, probajte napraviti što efikasnije implementacije odgovarajućih algoritama. Za rješenje smijete koristiti i dodatnu literaturu, uključivo i programe s weba (“sve što nađete”), samo propisno citirajte. Ako koristite “tuđe” algoritme, budite oprezni i pogledajte/testirajte ih prije korištenja!

Primjer jednostavnog mjerenja vremena (`dsecnd`) za Hanojske tornjeve je dostupan na webu. Smijete koristiti i bolju štopericu, ako ju nađete (molim, pošaljite mi ju, za buduće generacije).

Uputa za izravne linkove na literaturu. Početak (prefiks) pune adrese za **sve** članke je

`http://degiorgi.math.hr/oa/`

Na to treba dodati, tj. konkatenerati relativnu adresu (sufiks) za svaki pojedini članak. Tako dobivenu cijelu adresu treba kopirati u web preglednik i onda spremiti članak.

Zadatak 1. Brzo računanje n -tog člana rekurzije.

- Napišite program za brzo računanje n -tog člana niza zadanog k -članom homogenom rekurzijom s konstantnim koeficijentima. Uzmite da su koeficijenti u rekurziji i početni članovi nenegativni cijeli brojevi, tako da isto vrijedi i za sve članove niza. Program treba egzaktno izračunati n -ti član niza. Implementirajte operacije zbrajanja i množenja velikih brojeva u nekoj povoljno izabranoj bazi b (poželjno da je potencija od 10, za lakši ispis). Smijete koristiti i algoritme za brzo množenje velikih brojeva. Testirajte za rekurzije poput Fibonaccijevih brojeva, s relativno malim brojem k (ne treba preko 5) i dovoljno velikim n . Poželjno je da program radi barem za n u tisućama! Obratite pažnju na programsku realizaciju “rekurzivnih” dijelova algoritma, tako da eventualna alokacija/dealokacija memorije ne troši previše vremena. Posebno, pokušajte izbjeći nepotrebna kopiranja polja (nizova znamenki i slično)!

Literatura: predavanja, potražite po webu.

Zadatak 2. Mergesort algoritam za sortiranje polja ili liste.

- Napišite program koji uspoređuje razne varijante Mergesort algoritma za sortiranje polja, odnosno, vezane liste cijelih brojeva. Kod sortiranja polja, dozvoljeno je koristiti jedno dodatno polje. Kod sortiranja liste, dozvoljene su samo zamjene pokazivača. Razne varijante algoritma sastoje se u tome da polje/listu dijelimo na k dijelova podjednake duljine, s tim da k varira od 2 (standardni algoritam), pa sve do \sqrt{n} , gdje je n broj elemenata u polju/listi. Testirajte program za razne vrste polja/liste, variranjem duljine polja/liste i rasporeda elemenata u polju/listi. Smijete usporediti iste varijante algoritma na raznim strukturama: na polju i na listi. Također, smijete usporediti trajanje ovih algoritama s trajanjem `qsort` funkcije iz standardne C biblioteke.

Literatura: predavanja, predavanja iz Programiranja 2, dijelovi 5. poglavlja iz JS knjige (JS 219–225, dostupno na webu), CLRS 29–35, Knuth TAOCP 3 (javite se, ako želite), potražite po webu.

Zadatak 3. Minimalno razapinjuće stablo.

- Napišite program koji implementira Kruskalov algoritam za nalaženje minimalnog razapinjućeg stabla zadanog neusmjerenog grafa $G = (V, E)$ s pozitivnim težinama bridova. Iskoristite strukturu disjunktih skupova. Smijete usporediti i razne varijante za pojedine operacije u toj strukturi (pogledajte predavanja). Testirajte program za razne vrste grafova, variranjem broja vrhova i broja bridova (“gustoća”) grafa.

Literatura: predavanja, dijelovi 7. i 3. poglavlja iz JS knjige (JS 275–283, 284–295, treba još i 133–149, 150–162, dostupno na webu), KT (4.5, 4.6) 142–157 (dostupno na webu).

Zadatak 4. Par najbližih točaka u ravnini i više dimenzija.

- Zadan je niz (polje) od n točaka u ravnini, odnosno, k -dimenzionalnom prostoru. Svaka točka zadana je k -torkom koordinata (realnih brojeva). Napišite program koji uspoređuje algoritme za nalaženje najbližeg para točaka u zadanom nizu. Udaljenost točaka $d(x, y)$ je standardna Euklidska udaljenost, tj. $d(x, y) = \|y - x\|_2$. Možete koristiti algoritam iscrpne pretrage (ili “grube sile”) i razne brže algoritme — na primjer, bazirane na strategiji “podijeli–pa–vladaaj”. Testirajte program za razne nizove točaka iste duljine i razne duljine nizova (n varira).

Ideje za proširenje: smijete varirati i dimenziju k prostora (od 1 ili 2, do nekog razumno malog broja — ne treba ići preko 10), a onda koristite tzv. višedimenzionalni “podijeli–pa–vladaaj”. Također, možete napraviti sličnu usporedbu za druge izbore udaljenosti (na primjer, preko 1-norme ili ∞ -norme).

Literatura: dijelovi poglavlja o strategiji “podijeli–pa–vladaaj” u knjigama JS 225–232, CLRS (33.4) 834–840, KT (5.4) 225–231, ALS 192–195 (dio je dostupan na webu), za višedimenzionalni “podijeli–pa–vladaaj” pogledajte članak (1730 kB) na sljedećoj relativnoj adresi

`oaa_lit/k_dim_dc.pdf`

(str. 1–2, 11–15), potražite po webu.

Zadatak 5. Brzo množenje velikih brojeva.

- Napišite program koji uspoređuje trajanje standardnog i nekih “brzih” algoritama za množenje velikih prirodnih brojeva. Možete koristiti Karacubin algoritam i generalizacije iz familije brzih algoritama s parametrom r , za male vrijednosti r ($r \leq 5$). Obratite pažnju na realizaciju rekurzivnog dijela algoritma, tako da alokacija/dealokacija memorije ne troši previše vremena. Posebno, pokušajte izbjeći nepotrebna kopiranja blokova znamenki! Testirajte program za razne “duljine” brojeva. Možete zadatak proširiti i na dijeljenje velikih brojeva.

Literatura: predavanja, potražite po webu. Za dijeljenje pogledajte materijale iz Aritmetičkih i algebarskih algoritama (ili se javite), Knuth TAOCP 2, potražite po webu.