Napredni software Mathematica - Projekt

Algoritmi sortiranja i još ponešto...

Rok: Subota, 31.03.2012. godine

1 Osnovne informacije

- Vaš prvi dio ispitivanja iz ovog predmeta će se odnositi na jedan kratki izvještaj o vašem ispitivanju rada algoritama sortiranja u prvom, te dodatno konkretnog programiranja matematičkog programa u drugom dijelu.
- Projekat predajete u pismenoj i elektronskoj formi - izvještaj na papiru, dok programski kod i rad ili elektronskim putem (na email vedad.pasic@untz.ba) ili na disku ili USB memoriji.
- Rok za predaju rada (najkasniji) je subota, 31. mart 2012. godine u 12 sati. Predaja projekta poslije ovog roka vodi do automatskog gubitka 50% dodjeljene ocjene.
- Svaki oblik plagiranja, bilo jednih od drugih ili resursa sa Interneta će dovesti do **neizmjerno ozbiljnih posljedica** – svatko je nevin dok se ne dokaže suprotno dakako, no ne igrajte se sad pred kraj svojih studija! Slobodno radite skupa, no ne kopirajte jedni od drugih rad i navedite ukoliko ste radili u grupi i s kim.
- NE PLAŠITE SE! Ovaj projekat je namjerno napisan obimno kako bi vam POMOGAO, ne kako bi vam dao previše posla.
- Samostalni rad je mnogo ugodniji način dobivanja ocjene od ispita, koji u ovakvom predmetu nema ni puno smisla.

2 Potrebni dijelovi projekta

2.1 Implementacija algoritama sortiranja

Implementirajte u programskom jeziku Mathematica algoritme sortiranja i to:

1. Sortiranje umetanjem;
2. Bubble–sort; (vidi pseudo-code ispod)
3. Merge–sort;
4. Quick–sort.

Svaki algoritam možete implementirati na bilo koji način koji prati načelno pseudo–kod dat na predavanjima za ove algoritme.

Svaki algoritam takodjer možete implementirati na više načina – uz objašnjenje koje su fundamentalne razlike i da li tim promjenama algoritam postaje bolji ili gori i zašto?

Posebnu pažnju ovdje obratite algoritmu Bubble–sort koji nismo pokrili na predavanjima.

OBAVEZNO KOMENTIRAJTE PROGRAMSKI KOD!

Ocjenjivač mora biti u mogućnosti pročitati vaš kod i razumjeti šta ste htjeli uraditi.

2.2 Testiranje algoritama sortiranja

Svrha ovog projekta je da sami shvatite osnovne razlike izmedju različitih implementacija fundamentalno istog algoritma, i koliko 'pametno' programiranje može život učiniti lakšim. Algoritmi sortiranja su samo jedan primjer ovoga, dovoljno dobar za ilustraciju u edukativnom ciklusu.

U ovom dijelu morate *sami* smisliti način testiranja ovih algoritama, koristeći npr. funkcije **Timing** i **Random**, te sve što ste naučili u ovom kursu – to npr. može (a ne mora) da uključuje:

- Stvaranje nasumičnih listi koje ćete sortirati;
- Kako ćete svaki implementirani algoritam testirati na kreiranim podatcima;
- Kako ćete modifikovati vaše algoritme da rade još bolje;
- Kako prikazati grafički rezultate vaših testova;

- Kako različite vrste podataka mogu uticati na brzine rada različitih algoritama (npr. sortirana/nesortirana lista i drugo);
- Kako se brzina rada algoritma ponaša sa porastom broja elemenata u listi koju sortirate;
- Kolika je veličina listi koje testirate;
- Statistički uporediti rezultate (?);
- itd. itd. itd. ...

Gornja lista *nije* lista obaveznih stvari koje morate uraditi u vašem testiranju! Ona je samo tu da bi vam dala ideje kako da koncipirate svoju tehniku testiranja i kako da protumačite rezultate vašeg testiranja. No samo testiranje je u potpunosti ostavljeno *vama*.

2.3 Program ispitivanje toka i crtanja grafa funkcije

Koristeći se funkcijama *Mathematica*-e napraviti funkciju $MojaFunkcija[f\_, x\_]$ koja za proizvoljno unešenu funkciju $f(x)$ ispituje deset osobina funkcije

1. Domen funkcije.
2. Parnost funkcije. Periodičnost funkcije.
3. Asimptote funkcije.
4. Tačke presjeka grafika funkcije sa koordinatnim osama.
5. Znak funkcije.
6. Tok funkcije.
7. Ekstremi funkcije.
8. Intervali konveksnosti i konkavnosti funkcije.
9. Prevojne tačke funkcije.
10. Grafik funkcije.

Za ispitivanje navedenih osobina koristiti što je moguće vise funkcija softvera *Mathematica* ali je poželjno i da se naprave vlastite funkcije. Za output koristiti se funkcijom *Print* i poruke otprilike trebaju biti oblika:

- Funkcija $f(x) = \ldots$ ima (nema) vertikalnu asimptotu $\ldots$. Funkcija $f(x) = \ldots$ ima (nema) horizontalnu asimptotu $\ldots$. Funkcija $f(x) = \ldots$ ima (nema) kosu asimptotu $\ldots$.
- Presječne tačke grafika funkcije $f(x) = \ldots$ i koordinatnih osa su $\ldots$.
- Funkcija $f(x) = \ldots$ ima dostiže maksimum $y_{\max} = \ldots$ za $x = \ldots$.
- $\ldots$.

Prilikom crtanja grafika funkcije, ostaviti mogućnost izbora prikaza grafika funkcije sa/bez asimptota funkcije.

2.4 Izvještaj

Morate napisati i predati *kratki* izvještaj o vašem radu napisan ili u LaTeX – u ili u nekom Microsoftovom paketu (kao što je npr. MS Word), koji bi minimalno trebao sadržavati:

- Objašnjenje implementacije algoritama;
- Objašnjenje tehnike testiranja;
- Usporedbu rezultata testiranja i vaše zaključke;
- Sav programski kod u dodatku - kako implementaciju algoritama, tako i testiranje.
- Elektronsku formu vaše implementacije i testiranja.

2.5 Finalno

Puno sreće i koristite vježbe za rad na vašem projektu, te ukoliko niste u prilici raditi kod kuće, obratite se meni ili predmetnom asistentu kako bi vam omogućili pristup računarskom centru. I pitajte, pitajte, *pitajte*!

SRETNO!

Prvi pseudo-kod algoritma Bubble–sort

```
Algoritam bubbleSort(A)
    Input: Lista A dužine n
    Output: Sortirana lista
    zamjena = True
    Ponavljaj dok je zamjena istinita
      zamjena = False
      Za sve elemente liste A uradi
        Ako su susjedni članovi liste u suprotnom poretku
          Zamjeni poredak tih članova liste
          zamjena = True
    vrati sortiranu listu
```

Drugi pseudo-kod algoritma Bubble–sort

```
Algoritam bubbleSort(A)
    Input: Lista A dužine n
    Output: Sortirana lista
    Za sve elemente liste A od 1 do n
      sortirano = True
      Za sve elemente liste A od 1 do n-i
        Ako su susjedni članovi liste u suprotnom poretku
          Zamjeni poredak tih članova liste
          sortirano = False
        end If
      Ako je sortirano = True, izadji iz petlje i vrati sortiranu listu
```