

Ohjelmointi 1 / syksy 2007

17/20: Keittokirja IV

Paavo Nieminen

nieminen@jyu.fi

Tietotekniikan laitos

Informaatioteknologian tiedekunta

Jyväskylän yliopisto

Tavoitteita ennen kurssin loppua

- Demovastausten hutkiminen: oikaistaan yleisiä virheitä ja väärinymmärryksiä. Kantapää on hyvä oppimisväline.
- Tärkeimpien asioiden kertaamista: Viimeisillä luennoilla ehtii käymään uudelleen läpi asioita, jotka tuottavat vaikeuksia. Ehdotuksia kannattaa esittää! Eniten ehdotetut toteutuvat.
- Koodauskäytänteissä loppuun asti: sano mitä tarkoitat (Eli tehdään saman tien hyvää koodia eikä ruttuista spagettia)

Tämän luennon rakenne

Eka sessio:

- Viime viikon keittokirjasta jäi reseptejä tekemättä; jatketaan sen minkä ehtii.

Toka sessio:

- Välipala: Käytännön taitona binääri/heksa/oktaali-muunnokset etumerkittömille kokonaisluvuille
- Välipala: Käytännön taitona yksinkertaisen syntaksin määrittelemineen; syntaksikäsitteen sovelluksia.
- Alustavasti "Oppikirja-algoritmeja": yksinkertainen lajittelu, sanojen esiintymistiheyden analysointi

Tehtäväkuvauksesta ohjelmaksi 6

- Tehdään alusta alkaen kirjoitusnopeustesti.
- Tehtäväkuvauksen saaminen, "domain analysis"
 - Algoritmin miettiminen; rajapintojen miettiminen
 - Jakaminen alialgoritmeiksi; näiden rajapinnat
 - Algoritmien toteuttaminen yksi kerrallaan.

Koetetaanpa ympätä tähän keissiin tuollainen aihe:

- Dynaamisen eli vaihtuvakokoisen tietorakenteen idea: `ArrayList<AlkioidenTyyppi>`

Tehtäväkuvauksesta ohjelmaksi 7

→ Katsotaan, miten tehtäisiin Mandelbrotin joukon visualisointi. Tässä joutui sen verran pähkäilemään, ettei ihan luennon aikana onnistu toistamaan, mutta todetaan samat vaiheet kuin aina ja katsotaan koodi kehitysvaiheineen läpi.

- Tehtäväkuvauksen saaminen, "domain analysis"
- Algoritmin miettiminen; rajapintojen miettiminen
- Jakaminen alialgoritmeiksi
- Algoritmien toteuttaminen yksi kerrallaan.

Binääriluvusta heksaksi tai oktaaliksi

– helppo homma!

- Keksitään etumerkitön binääriluku, ja muutetaan se heksaluvuksi
- Muutetaan se oktaaliluvuksi
- Toisin päin: Keksitään hekso- ja oktaaliluvut, ja muutetaan ne etumerkittömäksi binääriluvuksi.

Etumerkillisten binäärilukujen esitysmuoto (ns. kahden komplementtiesitys) tulee jatkokursseilla tai itseopiskellen!

Syntaksin määrittäminen: miten ja erityisesti miksi?

→ Määritellään syntaksi merkkijonolle, jolla voi kertoa säätilan (sataa/paistaa aurinko/on pilvipouta) seuraavissa paikoissa: Agora, Jyväskylä, Vaajakoski. Semanttisia tarkistuksia ei tarvitse miettiä, siis Agorassa voi sekä sataa että paistaa; esim:

Agorassa sataa; Jyväskylässä on pilvipouta; Jyväskylässä paistaa aurinko; Agorassa paistaa aurinko

Miksi syntaksiasioita mietitään ohj1:llä (vain pari esimerkkiä):

1. Kyky määritellä syntaksi implikoi kykyä ymmärtää jonkun toisen kirjoittama syntaksi – esimerkiksi ohjelmointikieli
2. Esim. kätevän printf-tulostuksen ensimmäinen parametri on tietyn syntaksin mukainen merkkijono (→ katsotaan dokumentaatiosta kohta, jossa syntaksi on määritelty).
3. Esim. kätevien Regular Expression -työkalujen käyttäminen on olennaisesti yksinkertaisen syntaksin määrittelyä.
4. Tietokoneessa esitetään monesti kaikenlaista tietoa jonkun syntaksin mukaisena merkkijonona. Syntaksin määrittely (tai olemassaolevan syntaksin valitseminen) voi olla osa sovellusohjelman tekemistä. Esim. HTML-tiedostot, erilaiset XML-formaatit, ... (Ohjelmointi 1 on pohja monille jatkokursseille, joissa asiaan syvennyttään; kannattaa siis jo alustavasti miettiä.)

”Oppikirja-algoritmeja”

Jokohan voimme palata ensimmäisen luennon esitysmuotoon: kalvot leikkimielisenä Java-koodina?

```
luento = WorldOfAgora.getCurrentLecture();
try{
    luento.teachSubject("Yksinkertainen lajittelu");
    luento.teachSubject("Sanojen esiintymistiheys"
        + " - avuksi luokka HashMap");
} catch (LectureOutOfTimeException e) {
    luento.postponeRemainingUntilTomorrow();
} finally {
    everybody.toLunch();
}
```

... Siis yritetään käydä jotain läpi, mutta jos yllättävästi aika loppuu kesken, siirretään käsittely huomiseen.

Alkioiden lajittelu

(Tehtäväkuvauksesta ohjelmaksi 8)

- Toteuta kokonaislukutaulukon lajittelu.
- Tehtäväkuvauksen saaminen, "domain analysis"
 - Algoritmin miettiminen; rajapintojen miettiminen
 - Jakaminen alialgoritmeiksi
 - Algoritmien toteuttaminen yksi kerrallaan.

Tehdään nyt tosi yksinkertainen lajittelu – kunhan saadaan joku ratkaisu. Esim. oppikirjassamme on esitelty useita vaihtoehtoisia lajittelualgoritmeja luvussa "10 Kokoelmat".

Sanojen esiintymistiheys tekstissä (Tehtäväkuvauksesta ohjelmaksi 9)

- Laske sanojen esiintymistiheys merkkijonossa.
- Tehtäväkuvauksen saaminen, "domain analysis"
 - Algoritmin miettiminen; rajapintojen miettiminen
 - Jakaminen alialgoritmeiksi
 - Algoritmien toteuttaminen yksi kerrallaan.

(Java-alusta tarjoaa ruohonjuuritason välineet tähän ja moneen muuhun! Sovellus itse on kuitenkin karkean tason algoritmi, ja sen tekeminen edellyttää tietoa välineiden olemassaolosta ja taitoa käyttää niitä)