

Helsingin yliopisto/tietojenkäsittelytieteen laitos
Tietokannan hallinta / kurssikoe 25.6.2003 / H.Laine

Kirjoita jokaiseen erilliseen vastauspaperiin kurssin nimi, tenttipäivämäärä, nimikirjoituksesi ja nimen selvennys sekä syntymäaikasi.

Anna vastaukset kahdella konseptilla siten että (1&2) ja (3&4) ovat yhdessä.

Ensimmäisellä vastauskonseptilla:

1. Selitä lyhyesti RAID levyjen toimintaperiaate sekä levyjen hyödyt/haitat. (12)

[Katso kalvomateriaali.](#)

2. a) Selvitä B+ -puun lisäysalgoritmin periaate.

[Katso kalvomateriaali.](#)

b) Montako tietuetta enintään ja vähintään voidaan tallettaa B+ -puuhun, jonka korkeus on 4, kun puun lehtisivulle mahtuu enintään 20 tietuetta ja hakemistosivulle enintään 100 osoitinta.

[Maksimi= \$100 \cdot 100 \cdot 100 \cdot 20 = 20\,000\,000\$](#)

[Minimi= \$2 \cdot 50 \cdot 50 \cdot 10 = 50\,000\$](#)

(12)

Toisella vastauskonseptilla:

3. Tarkastellaan tauluja Tilaus(tunnus, tilauspäivä, toimitusosoite, asiakasnumero, muuta) ja Asiakas(asiakasnumero, nimi, paikkakunta, laskutusosoite, muuta). Taulussa tilaus on 30000 riviä, joiden keskipituus on 400 tavua, ja taulussa asiakas 3000 riviä, joiden keskipituus on 200 tavua. Käytettävällä levyllä sivukoko on 4KB. Mikä on tehokkain tapa toteuttaa kysely

```
select tunnus, nimi, laskutusosoite, paikkakunta
from tilaus, asiakas
where tilaus.asiakasnumero=asiakas.asiakasnumero;
```

kun käytettävissä on puskuuri-/keskusmuistitilaa tilaa noin 200 sivun verran (voit tehdä arviointia helpottavia oletuksia). Taulut on toteutettu kasarakenteena. Oletetaan että tauluilla on indeksit pääavaimen perusteella. Taululla tilausnumero on lisäksi indeksi myös asiakasnumeron perusteella. Indeksit on toteutettu hajautusta käyttäen. Ylivuotosivuja on indekseissä niin vähän, ettei niitä tarvitse huomioida. Perustele vastauksesi. (14)

[Tässä ollaan tekemässä liitosta asiakasnumeron perusteella. Jokaisella tilauksella on asiakas, joten tuloksen koko on 30000 riviä. Tuloksen kirjoittamiseen tarvittavia levyoperaatioita ei seuraavassa lasketa.](#)

Lasketaan aluksi taulujen koot:

- Tilaus: enintään 10 riviä sivulla, sivuja vähintään 3000. Koska tilaukset eivät muutu voidaan käyttää täyttösuhdetta 100%.
- Asiakas: sivulla enintään 20 riviä. Sivuja vähintään 150. Yksinkertaistusoletus: täyttösuhde 100%.

Jokainen tilausrivi pitää välttämättä lukea. Asiakkaista on pakko lukea vain ne rivit, joihin liittyy tilauksia.

Levyhakujen kannalta tehokkaita ratkaisuja ovat ne, joissa taulut käydään vain kertaalleen läpi, eli levyhakuja tulee hakemiseen $3000+150$. Tällaisia ovat:

- **Jaksotetut sisäkkäiset silmukat** kun Asiakas-taulu on ulompi ja luetaan kokonaan muistiin. Tällöin jää Tilausten lukemiseen 50 puskurisivua, joten sen hakemisessa voidaan hyödyntää peräkkäisyyttä.
- **Hajautusliitos** toimii samalla hakumäärällä. Oletetaan, että asiakasnumerot ovat järjestysnumeroita. Tällöin on määriteltävissä hajautin, joka ei tuota ylivuotosivuja. Hajautukseen riittää 150 sivun keskusmuistialue ja tilaustaulua voidaan käsitellä kuten jaksotettujen sisäkkäisten silmukoiden yhteydessä.

Indeksiliitoksessakin on jokainen tilausrivi luettava. Jos Asiakas-taulu on ulompi ja indeksihaku tehdään tilaustauluun, joudutaan sama sivu todennäköisesti hakemaan useasti (puskuritilaa jää vain alle 100 sivua koska indeksi vaatii noin 100 sivua). Hakuja teoriassa $150 + 4000*7,5 * +100$, mutta käytännössä $150 + n*3000 +100$, $1 < n < 10$.

Jos **tilaustaulu on ulompi**, niin olisi mahdollista selvittää jopa alle 3150 levyhaun, mikäli tilukset keskittyisivät vain muutamalle asiakkaalle. Koska puskuritila riittää asiakastaulun Sivujen pitämiseen keskusmuistissa on hakujen määrä $< 3000 + n \cdot 10$ (indeksisivut) $+150$, eli pahimmassa tapauksessa hieman enemmän kuin 3160.

Lomitusliitos ylittää ainakin noin kolminkertaisesti edellä mainitun levyhakujen määrän koska tilaustaulu on järjestettävä (jonojen teko ja yksi lomituskierrös).

Yhteenveto: jaksotetut sisäkkäiset silmukat hajautusliitos ja indeksihaku käyttäen ulompana tauluna tilaustaulua ovat likimain yhtä tehokkaita.

4. a) Mikä on luku- ja kirjoituslukon ero?
b) Selitä ankan kaksivaiheisen lukituksen (strict 2 phase locking, 2PL) periaate.
c) Mitä hyviä ja huonoja puolia ankan kaksivaiheisen lukituksen käytöstä on? (12)
[katso kalvomateriaali](#)

Erikseen sähköpostitse:

5. **Laskuharjoitukset korvaava kotikoetehtävä.** Tehtävän vastaavat vain ne, jotka *haluavat korvata* siitä saatavilla pisteillä laskuharjoituspisteensä. Jätetyn vastauksen tuottama pistemäärä otetaan arvosteluun laskuharjoituspisteiden asemasta. Laskuharjoituspisteet näkyvät kurssin kotisivulla. Vastaukset on toimitettava sähköpostitse osoitteeseen Harri.Laine@cs.helsinki.fi torstaihin 3.7. klo 16 mennessä. Vastausmuotona käy ascii-tiedosto, MS Word:n doc-tiedosto, html, pdf tai ps tiedosto. Sähköpostitse voi lähettää myös pelkästään [url:n](#).

Tehtävä: Kuvaa IBM:n VSAM (Virtual Storage Access Method) KSDS (key sequenced data set) tiedostorakenne ja vertaa sitä B+ -puuhun. (10)