HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos
PL 68
00014 HELSINGIN YLIOPISTO

581330-2 Tietorakenteet

Toinen kurssikoe 16.12.2004

Kirjoita jokaisen vastauspaperin yläreunaan nimi, henkilötunnus tai opiskelijanumero, nimikirjoitus, kurssin nimi ja tenttipäivämäärä.

HUOM1: Jokainen tehtävä omalle arkilleen

HUOM2: Pseudokoodi ilman kommentointia ja toimintaperiaatteen esittelyä ei tuo pisteitä

HUOM3: Arvaus ilman perusteluja ei ole aikavaativuuden analysointia

1. (5p) *Hajautus*
   (a) Oletetaan että hajautettavat avaimet ovat *sosiaaliturvatunnuksia* ja hajautustaulun koko on $n \times 10000$ missä n on positiivinen kokonaisluku jonka suuruus vaihtelee ohjelman suoritusaikana. Valitse tilanteeseen sopiva hajautusfunktio. Kuvaile miten valitsemasi hajautusfunktio toimii ja perustele valinnan sopivuus.
   (b) Mitä tarkoitetaan hajautuksen yhteydessä *yhteentörmäyksellä*? Esittele jokin tapa miten yhteentörmäykset voidaan ratkaista. Hahmottele miten hajautusrakenteen operaatiot voidaan toteuttaa tässä tapauksessa ja mikä on operaatioiden vaativuus.
2. (9p) *Minimikeko*
   (a) selitä miten keko on talletettuna muistiin
   (b) mitä tarkoitetaan *keko-ehdolla*?
   (c) piirrä minimikeko jossa alkiot 5, 3, 17, 10, 84, 19, 6, 22 ja 9
   (d) suunnittele operaatio change-key(i,k) joka muuttaa keon kohtaan i talletettuun alkion avaimeksi k ja samalla pitää huolta keko-ehdon voimassa säilymisestä
   (e) analysoi operaation aika- ja tila-vaativuus
   (f) simuloi (d)-kohdan operaatiota kohdan (c) kekoon siten että alkion 84 arvo muuttuu ensin arvoksi 11 ja tämän jälkeen alkion 3 arvo muuttuu arvoksi 55
3. (6p) Suunnittele algoritmi seuraavaan tilanteeseen:
   - syötteenä yhteen tai kahteen suuntaan linkitetty lista (saat itse valita kumpaa listatyyppiä käytät) jonka solmuihin on talletettu kokonaislukuja
   - operaatio palauttaa (yhteen tai kahteen suuntaan) linkitetyn listan missä samat luvut ovat suuruusjärjestyksessä

   Analysoi algoritmin aika ja tilavaativuus. Algoritmisi ei tulisi toimia hitaammin kuin ajassa $\mathcal{O}(l^2)$, missä l on listan pituus.
4. (5p) Suunnittele algoritmi joka saa syötteekseen suunnatun verkon $G = (V, E)$ ja kaksi verkon solmua $s1$ ja $s2$. Algoritmi selvittää onko olemassa polkua solmusta $s1$ solmuun $s2$. Jos polku löytyy, algoritmi tulostaa jonkun solmut yhdistävistä poluista (polkuja siis voi olla useita mutta yksi riittää).

   Analysoi algoritmin aika- ja tilavaativuus.