

TKT20001 Tietorakenteet ja algoritmit (kevät 2018)

Kurssikoe 1 (5.3.2018)

Tentissä saa olla mukana käsin kirjoitettu yksi A4-kokoinen "luntilappu", jonka molemmilla puolilla saa olla tekstiä.

Vastaa kuhunkin tehtävään erilliselle konseptipaperille. Kirjoita jokaisen paperin yläkulmaan **kurssin nimi, kokeen päivämäärä, nimi, nimikirjoitus ja opiskelijanumero**. Vaikka jättäisit johonkin tehtävään vastaamatta, palauta silti vastauspaperi kyseiseen tehtävään.

Tehtävissä, joissa pyydetään algoritmia, voit käyttää luentojen (Cormenin) tyyppistä pseudokoodia tai muita ymmärrettäviä pseudokoodityylejä tai oikeaa ohjelmointikieltä, esim. Javaa. Jos käytät oikeaa ohjelmointikieltä, selitä erityisen hyvin, mitä ohjelmassasi tapahtuu, äläkä käytä mitään kielen erikoista piirrettä tai valmiita kirjastoja.

Vastaa kaikkien kysymysten kaikkiin kohtiin. Kokeen maksimipistemäärä on 22.

1. [4 pistettä]

(a) Pitääkö paikkansa, että $4n^3 + n + 7 = \mathcal{O}(n^3)$?

(b) Pitääkö paikkansa, että $2^n = \Omega(3^n)$?

Perustele vastauksesi täsmällisesti \mathcal{O} - ja Ω -merkinnän määritelmästä lähtien.

2. [6 pistettä] Kurssilla on esitetty mm. algoritmit (a) lisäysjärjestäminen, (b) lomitusjärjestäminen ja (c) pikajärjestäminen. Millaisiin tilanteeseen lisäysjärjestäminen sopii näistä parhaiten? Entä lomitusjärjestäminen? Pikajärjestäminen? Perustele lyhyesti käyttäen algoritmien aikavaativuuksia tai muita kurssilla esitettyjä ominaisuuksia.

Käännä!

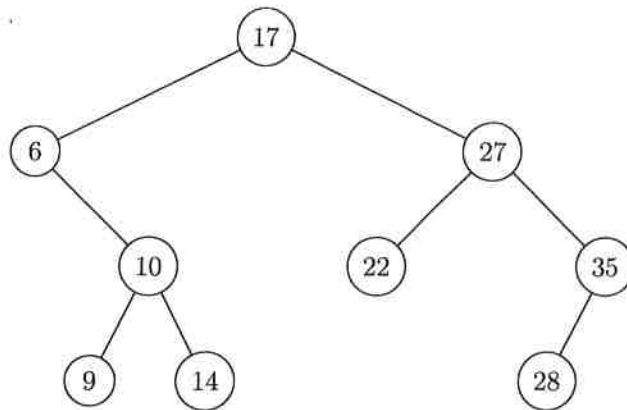
3. [6 pistettä] Syötteenä on annettu n -alkioinen taulukko, jossa sama arvo saattaa esiintyä useita kertoja. Tehtävänä on laskea, kuinka monta eri arvoa taulukossa esiintyy. Esitä tehokas ratkaisualgoritmi tehtävään kolmessa eri tilanteessa:

- (a) Taulukon alkiot ovat kokonaislukuja väliltä $[1, k]$, missä $k \leq n$.
Algoritmin pitäisi olla sekä aika- että tilavaativuudeltaan $\mathcal{O}(n)$.
- (b) Taulukon alkiot ovat kokonaislukuja, joille ei ole annettu mitään suuruusrajoja. (Kuitenkin oletetaan, että normaalit kokonaislukujen operaatiot voidaan suorittaa vakioajassa.)
Algoritmin pahimman tapauksen aikavaativuuden pitäisi olla $\mathcal{O}(n \log n)$ ja tilavaativuuden $\mathcal{O}(n)$.
- (c) Taulukon alkiot ovat merkkijonoja, jotka esittävät englanninkielisistä teksteistä löydettyjä sanoja.
Algoritmin keskimääräisen tapauksen aikavaativuuden pitäisi olla $\mathcal{O}(n)$, samoin tilavaativuuden.

Perustele algoritmiesi aika- ja tilavaativuusluokat. Voit käyttää apuna mitä tahansa kurssilla esitettyjä tietorakenteita, algoritmeja, aikavaativuuksia jne., kunhan selität, mitä tulosta milloinkin käytät. Ratkaisussa ei tarvitse esittää yksityiskohtaista pseudokoodia. Algoritmit riittää selittää sellaisella tarkkuustasolla, että aika- ja tilavaativuudet pystyy perustelemaan.

4. [6 pistettä] Syötteenä annetaan osoitin binäärihakupuun T juureen ja luonnollinen luku k . Tehtävänä on tulostaa puun T tasolla k olevat avaimet kasvavassa suuruusjärjestyksessä. Esitä yksityiskohtaisena pseudokoodina ratkaisualgoritmi tähän tehtävään. Algoritmin aika- ja tilavaativuutta ei tarvitse analysoida, mutta ratkaisun tulee olla kohtuullisen tehokas

Esimerkki Jos syötteenä annettu binäärihakupuun T on



ja $k = 2$, niin algoritmin pitäisi tulostaa luvut 10, 22 ja 35, tässä järjestyksessä.